(2)

TECHNICAL PAPER
CAA-TP-91-3

ALGORITHM FOR SITING POMCUS EQUIPMENT AT STORAGE FACILITIES (SITING)

SEPTEMBER 1991

PREPARED BY
FORCE SYSTEMS DIRECTORATE

US ARMY CONCEPTS ANALYSIS AGENCY
8120 WOODMONT AVENUE
BETHESDA, MARYLAND 20814-2797

CONCEPTS ANALYSIS AGENCY

DISCLAIMER

The findings of this report are not to be construed as an official Department of the Army position, policy, or decision unless so designated by other official documentation. Comments or suggestions should be addressed to:

Director
US Army Concepts Analysis Agency
ATTN: CSCA-FS
8120 Woodmont Avenue
Bethesda, MD 20814-2797

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved OMB No 0704-0188

1a. Report Security Classification: UNCLASSIFIED	1b. Restrictive Markings
2a. Security Classification Authority	3. Distribution/Availability of Report: UNCLASSIFIED/UNLIMITED
2b. Declassification/Downgrading Schedule	
4. Performing Organization Report Number(s): CAA-TP-91-3	5. Monitoring Organization Report Number(s)
6a. Name of Performing Organization: US Army Concepts Analysis Agency; 6b. Office Symbol (if applicable)	7a. Name of Monitoring Organization
6c. Address (City, State, and ZIP Code): 8120 Woodmont Avenue Bethesda, MD 20814-2797	7b. Address (City, State, and ZIP Code)
8a. Name of Funding/Sponsoring Organization: CAA; 8b. Office Symbol (if applicable)	9. Procurement Instrument Identification Number
8c. Address (City, State, and ZIP Code): Bethesda, MD 20814-2797	10. Source of Funding Numbers: PROGRAM ELEMENT NO / PROJECT NO / TASK NO / WORK UNIT ACCESSION NO

11. Title (Include Security Classification)
Algorithm for Siting POMCUS Equipment at Storage Facilities (SITING)

12. Personal Author(s)
Mr. Walter J. Bauman

13a. Type of Report	13b. Time Covered	14. Date of Report (Year, Month, Day)	15. Page Count
Final	From To	August 1991	172

16. Supplementary Notation

17. COSATI Codes

FIELD	GROUP	SUB-GROUP
15	05	01
12	04	

18. Subject Terms (Continue on reverse if necessary and identify by block number)
POMCUS, prepositioning, algorithm, deterministic, heuristic, logistics planning

19. Abstract (Continue on reverse if necessary and identify by block number)

Logistics planning for the European theater (and potentially for other theaters) provides for the prepositioned materiel (equipment) configured to unit sets (POMCUS) storage, such that the equipment is available in unit size clusters to be matched in theater with unit personnel in accordance with the theater operation plan (OPLAN). An algorithm has been developed to allocate the POMCUS to theater storage sites such that it is positioned as close as possible to General Defense Plan (GDP) positions where the linkup with unit personnel is to occur. The algorithm conducts a unit-by-unit, deterministic, heuristic search to locate the nearest (to the unit GDP position) storage facility which can store the set of equipment intact. The algorithm includes

20. Distribution/Availability of Abstract: ☐ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☒ DTIC USERS	21. Abstract Security Classification: UNCLASSIFIED
22a. Name of Responsible Individual: Mr Walter J. Bauman	22b. Telephone (Include Area Code): (301) 295-0308; 22c. Office Symbol: CSCA-FSC

Paragraph 19 (cont)

consideration of the OPLAN-specified priorities where the unit equipment may be stored. The storage allocation is made on year-by-year basis for each year of a multiyear planning cycle. Provision is made to minimize unit moves between years (turbulence). The algorithm is used as part of the POMCUS Unit Siting Alternatives (POMCUSITE) System. This system allows logistics planners to examine a broad range of POMCUS siting issues including equipment distribution, equipment storage (using the SITING algorithm), and the requirements for equipment movement to arrive at a preferred siting arrangement.

TECHNICAL PAPER
CAA-TP-91-3

ALGORITHM FOR SITING POMCUS EQUIPMENT AT STORAGE FACILITIES (SITING)

September 1991

Prepared by

FORCE SYSTEMS DIRECTORATE
Mr. Walter J. Bauman, Study Director

US Army Concepts Analysis Agency
8120 Woodmont Avenue
Bethesda, Maryland 20814-2797

This document was prepared as a technical reference to supplement the system information provided as part of a CAA study.

ALGORITHM FOR SITING POMCUS EQUIPMENT AT STORAGE FACILITIES (SITING)	STUDY SUMMARY CAA-TP-91-3

THE REASON FOR PERFORMING THE STUDY was to provide a baseline technical reference on the development of the allocation model denoted as SITING. The SITING model was developed in the POMCUSITE (POMCUS Unit Siting Alternatives) Study at the US Army Concepts Analysis Agency (CAA). SITING was designed to assist in POMCUS (prepositioning of materiel configured to unit sets) program management. This technical reference is needed to explain the full analytic nature and capabilities of the SITING Model, including features not used in the POMCUSITE Study.

THE STUDY SPONSOR is the Director, US Army Concepts Analysis Agency.

THE STUDY OBJECTIVES WERE TO document the analytic basis of the SITING Model, including inputs, outputs, and operation.

THE SCOPE OF THE STUDY addresses the efficient allocation of sets of equipment to storage sites in a POMCUS context over a timeframe of up to 8 consecutive years.

THE MAIN ASSUMPTIONS OF THIS WORK are:

(1) The use of Great Circle (GC) distances between allocation sites is appropriate for model applications.

(2) The penalty for moving a unit set between locations can be treated as proportional to the product of the unit set weight and the distance moved.

(3) Unit sets will not be fractionated (broken into pieces) in actual allocations.

(4) All (latitude, longitude) locations are in the northern hemisphere.

THE BASIC APPROACHES USED IN THIS STUDY were to:

(1) Define the allocation problem SITING is designed to solve.

(2) Describe the analytic nature of the allocation algorithm.

(3) Define the inputs and outputs of the model.

(4) Describe model operation and use.

THE PRINCIPAL FINDINGS of the work reported herein are:

(1) SITING is a deterministic allocation model written in FORTRAN for an IBM personal computer (PC). The model is designed to allocate sets of equipment (unit sets) efficiently over a set of storage sites.

(2) SITING allocates unit sets to storage sites while seeking to meet combinations of the following objectives:

(a) Each unit set is stored close to a predesignated (for that unit) location.

(b) The likelihood of an allocated unit set having to change storage site from one year to the next is kept small.

(c) Predesignated groupings of unit sets are usually allocated to a single storage site.

(3) The degree to which SITING achieves each of the above objectives is partly a function of user-specified options for SITING operation.

THE STUDY EFFORT was directed by Mr. Walter J. Bauman, Force Systems Directorate, US Army Concepts Analysis Agency.

COMMENTS AND QUESTIONS may be sent to the Director, US Army Concepts Analysis Agency, ATTN: CSCA-FSC, 8120 Woodmont Avenue, Bethesda, Maryland 20814-2797.

Tear-out copies of this synopsis are at back cover.

CONTENTS

FIGURES

TABLES

CHAPTER 1

GENERAL INFORMATION

1-1. PURPOSE. The purpose of this paper is to provide a standalone technical reference for the algorithm used for siting unit sets of equipment in storage locations reflecting proximity to points of operational need. The algorithm was developed for and incorporated* into the SITING module in the POMCUS Unit Siting Alternatives (POMCUSITE) Study (see CAA Study Report CAA-SR-91-8). A subset of this description has been adapted for inclusion in the POMCUSITE Study Report (Ref. 1), the POMCUSITE System User's Manual (Ref. 2), and the POMCUSITE System Documentation (Ref 3). All of the features and capabilities of the algorithm used in the POMCUSITE Study SITING Module are also included in the stand alone SITING Model described herein. However, the standalone SITING Model includes numerous features and capabilities that were deactivated in the POMCUSITE SITING Module because they were not required for POMCUSITE study objectives. A separate technical report on the standalone SITING Model serves to:

a. Describe the nature, use, and capabilities of a generalized version of the POMCUSITE SITING algorithm which can be used independently of POMCUSITE modeling architecture. (The POMCUSITE SITING algorithm was operationally embedded within a larger model which performed preprocessing and post-processing for the SITING Module.)

b. Describe features and capabilities of the algorithm which were not usable in the POMCUSITE algorithm, but which may prove useful for both further development of the POMCUSITE method as well as in other applications.

c. Describe the analytic approach used in the generalized SITING Model algorithm to the general modelling community so that it can be examined as a basis for use on allocation problems other than that addressed in the POMCUSITE Study. The SITING methodology applicability is not restricted to a European theater. The methodology can be applied to any problem involving prepositioned storage and can be extended to applications in a global theater.

The features of the SITING Model described herein which were not included in the POMCUSITE Study are summarized at the end of this chapter, after the basic problem addressed by the model has been defined.

1-2. VERIFICATION AND VALIDATION. The basic SITING algorithm was subjected to an independent verification and validation test procedure which was conducted external to the SITING Model design effort. That verification and validation design, as documented in the POMCUSITE Study Report, consisted of 18 test problems which exercised the SITING algorithm under a variety of scenario and algorithm conditions. The solutions generated by SITING were compared with anticipated outcomes predicted by the analyst from SITING design logic and scenario considerations. All test results conformed to anticipated outcomes.

*This paper includes algorithm features and capabilities which were deactivated in the fielded version of the SITING module to simplify the module operation.

1-3. BACKGROUND OF SITING ALLOCATION PROBLEM. Prepositioned equipment for each unit which will deploy early in wartime is stored under the concept of prepositioned materiel configured to unit sets (POMCUS). Individual unit sets are stored at POMCUS storage sites in the theater. These unit sets are grouped into "projects" with each project consisting of a cluster of unit sets. A unit set can be in only one project. In each year (of up to 8 consecutive years, beginning with the current year), the number and characteristics of POMCUS unit sets, projects, and storage sites may change, relative to the previous year. These changes will often force relocation of unit sets from their previous (year's) storage site to a new site (e.g., the unit sets which fit into a site last year may be too large to all fit there this year). The following management considerations arise:

a. Unit Set Positioning. In wartime, a unit set will have to be moved from its POMCUS storage site to a predesignated assembly area for the associated unit, denoted herein as a unit set general defense plan (GDP) position, from which the POMCUS equipment for the wartime owner of that unit set will be transferred to the owning unit during mobilization. It is desirable, in order to reduce delivery response time, that the movement of the unit set from its POMCUS storage site to its GDP be as short as possible. Thus, in an ideal environment, a POMCUS unit set would be stored at the POMCUS storage site which is closest to the unit set's GDP. **NOTE:** a common usage is that "GDP" means "GDP position."

b. Unit Set Resiting Turbulence. Moving equipment is costly. If a unit set stored at a site A in a year is to be stored at another site B during the following year, then the unit set is said to be "resited" (from site A to site B) in the second year. The frequency of occurrence of resiting over all unit sets allocated in a year is denoted as "resiting turbulence." It is desirable to have as little unit set resiting turbulence as possible in order to reduce management confusion. In an ideal environment, POMCUS unit sets can remain at their initial storage sites during the entire timeframe. (An environment in which all sites have unlimited storage capacity is an ideal one.) The existence of storage site capacity limits makes a nonideal environment likely.

c. Project Management. The unit sets are grouped into military operational clusters denoted as projects. The projects can be managed as a single entity. Therefore, all unit sets in a project can be issued together in an efficient manner. In an ideal environment, this can best be done if all unit sets in a project are stored at a single site. Since storage site capacity limits make this unlikely, the corresponding desirable goal is to store the unit sets belonging to a project at as few sites as possible. If the unit sets of a project are to be stored at separate sites A, B, and C, then that project's (stored) unit sets are said to be "dispersed over sites A, B, and C," and "reducing stored project dispersion" refers to reducing the number of sites at which the unit sets of each individual project are stored. The stated desirable goal is to reduce stored project dispersion for each project.

d. Unit Set/Project Priority Considerations. Each project can be assigned a numeric "project priority." Each unit set can also be assigned a numeric "unit priority." (A low value for numeric priority corresponds to high priority.) In a nonideal environment, allocation of individual projects

and/or unit sets to "best available sites" should usually be done in priority order so that higher priority units/projects have a higher probability of receiving preferred siting.

1-4. SITING ALLOCATION PROBLEM

a. The siting allocation problem is treated in the context of a formal algorithm in that it considers the allocation as a sequence of allocations occurring over a 2- to 8-year period. For each year, a separate allocation is carried out, following a prescribed sequence of processing which is based on the objective(s) of the processing to be sought and allocation rules which consider unit priorities, project priorities, and unit proximity to the operation plan (OPLAN) GDP. A detailed description of the algorithm is presented in Chapter 3.

b. In summary form, the algorithmic process is as follows. Over the (up to 8) years in the problem timeframe, designate the last year as the goal year. The first year is always the current year. Data on unit sets, project structure, and site characteristics are given for each year. The overall objective is to allocate all of the unit sets in a priority ordering (based on both unit set priority and project priority) in each and all years in terms of:

- **Objective 1.** Unit sets are stored (allocated) as close as possible to their OPLAN GDPs. (i.e., good unit set positioning is attained).
- **Objective 2.** The resiting turbulence (frequency of resiting of stored unit sets from year to year) is kept small.
- **Objective 3 (optional).** In addition to Objectives 1 and 2, the user may specify that unit set dispersal (over multiple sites) of stored (allocated) in the same project is reduced, according to the project priority of the owning project.

c. Objectives 1 and 2 are always operative in SITING allocations. Objective 3 is an optional addition and, if desired, is specified via interactive user input. Objective 1 treats unit set positioning considerations. Objective 2 treats unit set resiting considerations. Objective 3, if used, treats project management considerations. The use of unit set and project order during the allocation process addresses unit set/project priority considerations.

1-5. INVALID SITING ALLOCATION PROBLEM FORMULATION. It is possible that the user will give SITING a problem in which all of the available site storage space is less than (and therefore not enough to store) the total space requirement of all unit sets in the problem. Such a problem is not solvable and is therefore invalid. SITING will nonetheless try to generate a solution in this case by creating two notional "overflow sites," denoted as site -01 and site -99. Two sites are created because at most 200 units can be assigned to any one overflow site. Site -99 is used for up to 200 overflow units. If there is more overflow, site -01 is also used. These sites, being notional, have no designated coordinates. Their purpose is to serve as repositories for unit sets for which no room exists at any "actual" site. Any associated solution is invalid, but the allocations give useful information on the problem. When allocation overflow occurs, SITING alerts

the user, via on screen messages (see Appendix B), that overflow exists. To eliminate an overflow problem, the user must increase site capacities defined in input until no overflow occurs. Because SITING does not allocate pieces of unit sets, the total available floor space specified may well have to be slightly greater than the total of unit set storage areas in order to ensure a feasible nonoverflow solution.

1-6. SITING PROBLEM CONSTRAINTS AND LIMITS. Since a personal computer (PC) has limited memory, the types of problems solvable by it are constrained. These constraints are defined by setting certain program parameters in the SITING source code. Table 1-1 summarizes the chief SITING constraints along with the associated program parameter and range of values as defined in the SITING program source code. See Appendix B for a list of error messages associated with violation of these constraints during processing. A programmer familiar with the SITING code can easily alter these constraints by resetting program parameters in the SITING program source code.

Table 1-1. SITING Problem Constraints and Limits

1. Number of years in timeframe must be greater than 1 but less than 9.
2. Maximum number of different unit set identifications (IDs) is 1000 (over the entire timeframe).
3. Maximum value of a unit set numeric ID is 2000.
4. Maximum number of input storage sites over the entire timeframe is 28.
5. Maximum number of different project IDs is 25 (over the entire timeframe).
6. Maximum number of input designated unit set allocations is 50.
7. Maximum number of input designated project set allocations is 50.
8. Maximum number of unit sets initially in-place at any one site is 200.
9. Maximum number of unit sets that can be stored at any one site is 200.

1-7. ASSUMPTIONS. In addition to the constraints and limits described above, SITING logic includes the following assumptions:

a. All input data files are complete and are consistent with each other, e.g., each unit set is assigned to one and only one project.

b. No unit set can be stored "broken," i.e., fractions of a unit set cannot be allocated to different sites.

c. Unit set size (floor space required) and site capacity are expressed in the same units of measure.

d. The total storage space available at all sites is sufficient to accommodate all unit sets. (SITING does do a "workaround" invalid solution in this case.)

e. Distances between sites or between sites and GDPs are always by the shortest "straight line" great circle distances on a spherical earth.

f. Weight-distance is viable as a surrogate for response time; i.e., the penalty for moving a unit set between two locations (e.g., a storage site and a GDP) can be treated as proportional to the product of the unit set weight and the distance between the two sites.

g. All distances are in kilometers (km).

h. All (latitude, longitude) locations are in the same hemisphere.

1-8. SITING MODEL EXTENSIONS BEYOND POMCUSITE APPLICATIONS. The following comprise the main capabilities of the SITING Model documented herein which represent features unavailable in the algorithm employed in the SITING Module of the POMCUSITE Study.

a. The model can be executed as a standalone model. (The POMCUSITE algorithm was embedded in a larger model architecture encompassing specialized preprocessing and postprocessing.)

b. The problem timeframe length may be any integer value between 2 years and 8 years. (The POMCUSITE timeframe was "hard-wired" at 8 years.)

c. Measures of effectiveness (MOEs) which assess how well the algorithm solution achieves the problem objectives can be generated.

d. A solution which uses the fewest possible storage sites can be generated.

e. A solution can be generated based on changing all storage site capacities to an unlimited capacity. Such a solution gives guidance on storage site "sizing" which can achieve perfect accomplishment of problem objective 1.

f. The input in-place (starting) configuration of stored unit sets can be assessed, via the MOEs of 1-8c above, relative to the three problem objectives.

g. Selected unit sets and/or projects may be preallocated by user input.

h. More information about the solution is made available to the user in more output files.

Although this report describes the logic of the complete SITING Model, the main body concentrates on use of the "core" algorithm capabilities which also characterize the algorithm used in the POMCUSITE SITING module. Details on use of the extended capabilities, relative to the POMCUSITE Study application, are given primarily in Appendix E. The "core" algorithm is treated as a base standard version, and the extensions are treated as optional features, inputs, and outputs.

CHAPTER 2

SITING CONFIGURATION

2-1. HARDWARE REQUIREMENTS. SITING is configured for an IBM PC AT. It is a deterministic simulation written in Microsoft 5.0 FORTRAN. There is no need for Microsoft FORTRAN to be resident on the PC on which SITING is executed. In its current configuration, SITING execution requires 450K of random access memory (RAM) to execute. Approximately 1.5 megabytes of memory in a directory on the hard disc drive of the PC should be available to hold the SITING program (450K), the SITING input for the case processed (approximately 300K-400K for an 8-year case with 700 units), and the basic POMCUSITE output (approximately 500K).

2-2. INPUT REQUIREMENTS. SITING requires construction of four formatted ASCII data files. These must be resident, with the SITING program, on a directory on the hard disc drive of the PC at the time of SITING execution. There are two optional formatted input files which can provide the user with added problem flexibility in problem definition and output useful for solution analysis. The uses and capabilities of these optional inputs are described in Appendix E. However, these optional inputs are not linked, through data base interface, with the other POMCUSITE modules. A small amount of decision output selection is also entered by the user through the PC keyboard in response to onscreen menu prompts generated by SITING.

2-3. OUTPUT REQUIREMENTS. SITING writes formatted ASCII file output directly onto the hard disc drive directory on which the SITING program and the input data files are resident. Diagnostic message output is generated on the PC screen as required. No print output is generated by SITING.

2-4. SITING EXECUTION. To execute a case, the user must possess the SITING program disc, which contains the executable SITING program. To execute SITING for a case, the user must have previously loaded (copied) the SITING program disc into a directory on the PC. All of the input data for the case to be processed must also be resident in the directory containing the SITING program. At the disk operating system (DOS) prompt, while resident in the directory containing program and input data, the user enters (the command) SITING, and responds thereafter to interactive prompts from the screen. The user will then be prompted to specify a few scenario option inputs. Output of SITING will be written to the resident directory on the PC. The amount of output generated depends on the scenario. In tests, an 8-year case with 508 unit sets and 24 storage sites created 550K of basic output. The time required to process a case depends upon the scenario. Approximately 20 minutes of clock time on the PC was required for the test case with 508 unit sets. If a solution assessment (see Appendix E) is also desired, an additional 20 minutes is required. The test PC was a 286 version with a math coprocessor.

CHAPTER 3

SITING ALGORITHM

3-1. INTRODUCTION. This chapter explains the analytic procedures of the allocation algorithm used by the SITING Model. The problem addressed by the SITING algorithm is briefly summarized, as are the model inputs. The SITING allocation algorithm is described, and flow charts of algorithm logic are also presented.

3-2. THE SITING PROBLEM. The components of the problem are the units sets to be stored and the grouping of the unit sets into projects.

a. Unit Sets. Prepositioned equipment for each unit which will deploy in wartime is stored as a unit set at a (storage) site in theater. Upon wartime mobilization, each unit set must be transferred from its storage site to another location, associated with that unit, from which the equipment in the unit set will be picked up by its owner unit. This second pickup point is denoted herein as the unit general defense plan (GDP) location for that unit set.

b. Projects. Unit sets are administratively grouped into clusters denoted as "projects." Each project consists of a cluster of unit sets. A unit set can belong to only one project. In each year of the timeframe of up to 8 consecutive years, beginning with the current year, the number and characteristics of the stored unit sets, projects, and sites will change, relative to the previous year. Each project is assigned a numeric "project priority." Each unit set is assigned a numeric "unit priority" (a low value corresponds to high priority). Allocation of individual projects and/or unit sets to be stored at "best available sites" should usually be done in priority order so that higher priority units/projects have a higher probability of receiving preferred siting.

3-3. STATEMENT OF PROBLEM. Over all consecutive years in the timeframe, designate the last year as the goal (target) year. Year 1 is the current year. Data on unit sets, project structure, and site characteristics are given for each year. The objective is to allocate all of the unit sets in a priority ordering (based on unit set priority and project priority) in each and all years so that:

Objective 1. Unit sets are stored (allocated) as close as possible to their unit set GDPs (reduced site to GDP distance).

Objective 2. Unit sets are stored so that the resiting of stored unit sets between storage sites from year to year is kept small (reduced unit turbulence).

Objective 3. Optionally, the user may specify that unit set dispersal (over multiple storage sites) of unit sets in the same project is kept to a minimum (reducing project dispersion).

3-4. ASSESSMENT OF OBJECTIVES. SITING generates a solution siting plan. However, the user can use optional inputs, described in Appendix E, to have SITING also assess the generated solution relative to how well it achieves

the three objectives described in the problem statement. These optional SITING measures of effectiveness (MOEs) are summarized in Table 3-1 and are described in detail in Appendix E.

Table 3-1. SITING Solution Assessment MOEs

SITING MOE	What MOE measures
Avg wt x km for unit set moved from storage site to GDP	Objective 1 (positioning)
Fraction of unit sets changing storage site from previous yr	Objective 2 (resiting turbulence)
Number of sites over which each project is stored	Objective 3 (project dispersion)
Largest fraction of each project stored at a single site	Objective 3 (project dispersion)
Total number of sites used to store all unit sets	Objective 3 (project dispersion)

3-5. MODEL INPUTS

a. Data Inputs

(1) Unit Set Characteristics. For each year of the timeframe, the size (inside warehouse floor space required for storage), weight, unit set priority, and unit set GDP coordinates.

(2) Storage Site Characteristics. For each year of the timeframe, the total warehouse floor space capacity of each storage site and the location of each storage site.

(3) Project Characteristics. For each year of the timeframe, the project priority of each project and identification of the unit sets comprising each project.

(4) In-place Characteristics. The storage site location of each unit set at the start of the first year of the timeframe.

(5) Timeframe. Number of years (between 2 and 8) in the timeframe.

(6) Designated Project or Unit Set Allocations. This input is optional. If this input is specified, the user wishes to preallocate specific projects or units at specific sites so that the algorithm will allocate "around" them. If a project or unit set does not fit at its designated site, the algorithm allocates it normally, as if the designated allocations were never specified.

b. Onscreen Options. A choice (if applicable) from each of the following option queries is selected by the user via interactive onscreen menu:

(1) Allocation Scenario Options. The user chooses one of the following:

(a) Available Storage Case. Each storage site's capacity is as specified by input. SITING will determine siting plan allocations for each year of the timeframe.

(b) Unconstrained Storage Case. All storage sites have essentially unlimited storage capacity. SITING will determine siting plan allocations for each year of the timeframe. There is no resiting turbulence (Objective 2) in this case. Also, the resulting siting plan allocates every unit set to its best-positioned storage site (Objective 1). This is a benchmark "best case."

(c) Baseline Case. No siting plan allocations are to be determined. Instead, the "goodness" of the input in-place siting of year 1 assets is assessed in light of how closely objectives 1, 2, and 3 are met. The unit set characteristics are those defined by input for year 1.

(2) Project Dispersion Options. This option is not applicable when the baseline case allocation scenario option is chosen. When applicable, choices are:

(a) Ignore Dispersion. If this option is specified, then only Objectives 1 and 2 will be sought. Objective 3 will be ignored. In this case, unit sets will be stored as close as possible to unit GDPs and resiting turbulence in each year is kept as small as possible.

(b) Reduce Dispersion. If this option is specified, then Objectives 1, 2, and 3 will be sought. In this case, unit sets will be stored as close as possible to unit set GDPs, resiting turbulence in each year is kept as small as possible, and the stored unit sets of each individual project will be dispersed over as few sites as possible while simultaneously seeking the indicated objectives.

(3) Redundant Site Limit Option. This option is applicable only if the "available storage case" allocation scenario option is also in force. The redundant site option enables a user to have SITING generate a solution which is restricted to the k largest capacity sites, where the integer k is implicitly specified by the user. SITING internally calculates a value, M = the minimum number of input storage sites needed to store all of the unit sets in the problem. These sites are necessarily the M largest capacity sites in the problem. Sites other than these are then denoted as "redundant site" because they are not needed in a minimum site solution. The user must input how many redundant sites should be used in the generated solution. Therefore, a user specification of N redundant sites will generate a solution restricted to the k largest capacity sites, where k = M + N. When applicable, the user must specify the maximum number of the "redundant sites" which they will allow to be added to the M basic "sites allowed." If the user specifies N of the redundant sites to be allowed, SITING will take the N largest site capacities of the redundant sites and will transfer the associated sites from the

redundant sites list to the "allowed sites" list. The remaining redundant sites are then removed from the problem for that year. The user can choose to make all sites available at input-rated capacity by setting the "redundant sites allowed" to a value larger than the number of storage sites in the problem (a value of 99 suffices).

3-6. SEQUENCE OF PROCESSING

a. Rationale. The SITING algorithm initially determines a single-year solution which emphasizes Objective 1 for just the final year of the time-frame . This final-year solution then becomes a baseline goal. Solutions for other years are, in a sense, subsequently "fitted" to conform to the final-year solution. Conformity to the final-year solution reduces resiting turbulence and therefore promotes Objective 2. Tradeoffs against Objective 1 are limited since a unit set's siting in the solution for the final year should also be a good choice, if not always the best choice, in other years of the timeframe.

b. Sequence of Processing Years in Timeframe

(1) Goal Year. Initially, the goal (final) year is processed. The primary purpose of this initial goal year processing is to determine a "best" pairing of a storage site with each unit which will serve as an allocation goal in all other years processed. The site chosen to store a specific unit set in the goal year is denoted as the "goal site" for that unit set in all other years. These goal site allocations are chosen by treating only Objective 1 (unit set positioning) and, if applicable, Objective 3 of the SITING problem. Objective 2 is not considered in this initial processing of the goal year.

(2) Year 1. After the initial processing of the goal year has determined the goal sites for unit sets, SITING processes all other years of the timeframe in order. Year 1 is first. The input in-place storage is not used by SITING to determine allocations. As explained below, SITING processes year 1 and determines all unit set allocations for the SITING solution that year.

(3) Year 2. The next year processed is year 2. In processing year 2, the allocation solution of year 1 is treated as the in-place siting for year 2 and is used in the determination of the solution unit set allocations for that year.

(4) Succeeding Years. Each year after year 2 is then processed in turn to determine all unit set allocations for that year. In each such case, the allocation solution for the previous year becomes the in-place siting for that year. The final year processed is the goal year again. The solution algorithm and the solution for the goal year is the same whether it is processed initially or at the end. The primary reason the goal year is repeated is to enable easy generation of results in the proper yearly sequence.

NOTE: The current SITING algorithm processes forward in time from the first year of the timeframe. This sequence was devised in light of the need to meet the specific problem objectives described in paragraph 3-3 in a balanced

fashion. The sequence described works very well for the problem described. In the current algorithm, priority is explicitly given to the final year/goal year of the timeframe. For other, but similar, problems, redefinition of the goal year and/or of the algorithm processing sequence by year may be considered. For example, in the current problem, one alternative is to continuously process backward in time from the final year/goal year. The appropriateness of an alternative processing sequence should be examined against the characteristics of a particular problem, or alternative sequences should be tested.

c. Sequence of Processing Unit Sets for Each Year. For each year, SITING allocates each unit set in a sequence based on a combination of the unit set priority and the priority of the project to which that unit set belongs. This combination of priorities for a unit set is denoted as the combined unit (set) priority. The sequence of combined unit set priorities is constructed so that:

(1) Projects are allocated sequentially according to their project priority order.

(2) The unit sets in each project are allocated consecutively in SITING according to their unit priority order.

Each unit set will be assigned a combined unit set priority that is numerically sequenced to achieve (1) and (2) above. The combined unit set priorities are defined in terms of the input unit set priorities and the input project priorities as follows:

(a) Over all unit sets in all years find a smallest integer N such that 10^N exceeds the value of the largest input unit set priority.

(b) For each unit set i, with unit set priority Ui, define the combined unit set priority as $(10^N)(Pi) + Ui$, where Pi equals the project priority of the project to which unit set i belongs.

The above construction method for combined priorities was designed to allow the individual unit sets to be processed in the same sequence as described in (1) and (2) above. Since the scale of the original input unit priorities is altered in the computed combined priorities, the use of those combined priorities is restricted to ordinal sequencing and ranking. Construction of a combined priority was done in SITING because it combined two unit set attributes (unit priority, project priority) into one attribute which simplified programming of logic using the sequencing described in (1) and (2) above. Its use was also limited to ordinal ranking. The original input unit priorities are always available and accessible within the SITING program if needed.

3-7. ALGORITHM DESCRIPTION FOR "IGNORE DISPERSION" OPTION. The last year of the SITING problem timeframe is denoted herein as the goal year. In all SITING allocation processing, the model first processes (determines allocations in) the goal year. Next, it processes year 1 through year [goal-1] in that order, after which it repeats the processing of the goal year. The algorithm procedure is:

a. Processing of Goal Year. SITING initially processes the goal year of the timeframe in order to assign each unit set a goal site. A unit set's goal site is the storage site to which it is allocated by SITING during this processing of the goal year. The unit set allocations to sites done by SITING in the goal year comprise the SITING solution siting plan for that year. As will be explained, these goal site assignments are also used in generating solution allocations for every other year. The goal year allocations are done on the basis of best positioning relative to the unit set GDPs (Objective 1). The algorithm logic flow for the goal year is shown in Figure 3-1. Sections of the logic flowchart are referenced in brackets below, after the subparagraph which describes processing in that section. The sequence of processing is:

(1) If any designated project allocation input is present for this year, SITING attempts to allocate, in order of project priority, the input-specified projects to the input-specified sites. [The first two blocks of Figure 3-1.]

(2) If any designated unit set allocation input is present for this year, SITING attempts to allocate, in order of input specification, the input-specified unit sets to the input-specified sites if they have not already been allocated by (1). [Entry point 1 and the following block in Figure 3-1.]

(3) Each unallocated unit set is processed (allocated) according to increasing order of combined unit set priority (as defined in paragraph 3-6 above).

(4) Each unallocated unit set is processed as follows:

(a) If possible, the unit set is allocated to the storage site nearest its GDP.

(b) If no site has room for this unit set, it is allocated to the notional overflow site -99 or to overflow site -01 (used if site -99 overflows). [Entry point 2 through the end of chart in Figure 3-1.] In all cases, if a unit set can fit nowhere, SITING allocates it to this notional site. In such a case, the problem is infeasible, and the SITING solution is not valid. In order to generate a valid solution, SITING will then have to be rerun with sufficient extra storage capacity defined to allow all unit sets to be stored in "actual" sites.

(5) A numeric example (simplified) illustrating the processing for the ignore dispersion option in the goal year is shown in Appendix C (Example 1, page C-2).

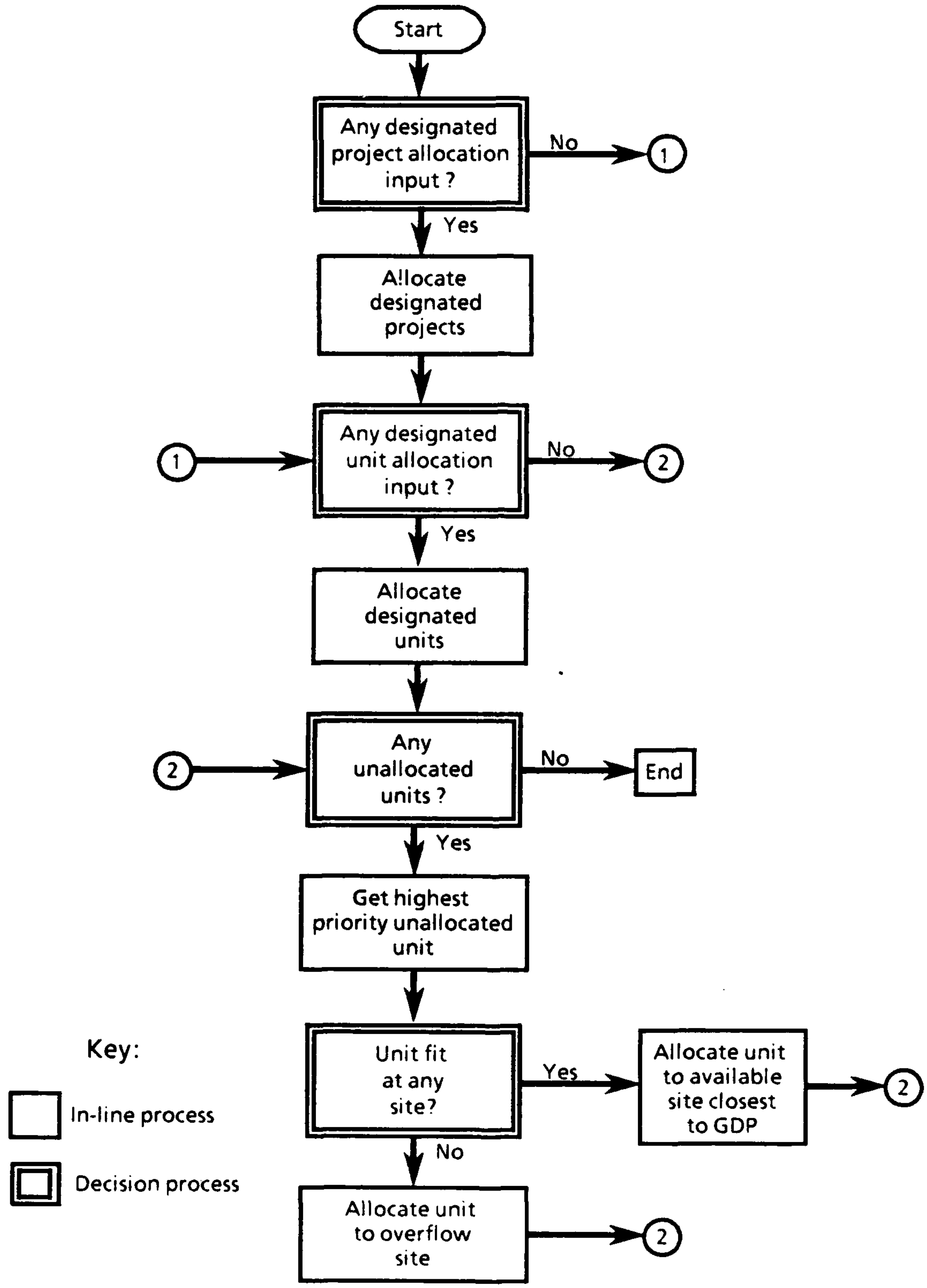

Figure 3-1. SITING Algorithm with Ignore Dispersion Option (goal year)

b. Processing of Year 1. Having processed the goal year, SITING continues by processing year 1 of the timeframe. The sequence of processing for year 1 is represented by the logic in Figure 3-2 with the following stages:

(1) All unit sets are assumed to be initially in place at their unit set goal sites, i.e., at their solution allocation sites in the goal year. (Any input in-place sites are ignored except for assessment purposes.)

(2) If any designated project allocation input is present for this year, SITING attempts to allocate, in order of project priority, the input-specified projects to the input-specified sites.

(3) If any designated unit set allocation input is present for this year, SITING attempts to allocate, in order of input specification, the input-specified unit sets to the input-specified sites.

(4) The remaining unallocated unit sets are processed in order of combined unit set priority.

(5) Each unallocated unit set is processed as follows:

(a) A unit set is allocated to its goal site, if possible.

(b) If this is not possible, the unit set is allocated to the site closest to its GDP which has space for it.

(c) If no site has space for it, the unit set is allocated to the overflow site (site -99).

A numeric example (simplified) illustrating the processing for the ignore dispersion option in year 1 is shown in Appendix C (Example 2, page C-5).

c. Allocation Processing for Other Years. Allocation processing of the years between year 1 and the goal year, but not including either, is as follows. The logic flow is also portrayed in Figure 3-2. Sections of the logic flow chart are referenced in brackets below, after the subparagraph which describes processing in that section. The logic of Figure 3-2 also applies to year 1 with the assumption that the goal year allocations (goal sites) are the in-place site locations in year 1. Processing in each of these years is as follows:

(1) If any designated project allocation input is present for this year, SITING attempts to allocate, in order of project priority, the input-specified projects to the input-specified sites [first two blocks of Figure 3-2].

(2) If any designated unit set allocation input is present for this year, SITING attempts to allocate, in order of input specification, the input-specified unit sets to the input-specified sites if they have not already been allocated in (1) [entry point 1 and the following block in Figure 3-2].

(3) The SITING allocations of the previous year become the "in-place" site locations of the unit sets in the year being processed. Unit sets appearing for the first time will not have an in-place site.

(4) All unallocated unit sets that are in-place at their goal sites are processed ("looked at") in this step, but not all are necessarily allocated in it. Each such unit set is processed as follows in order of combined priority [entry point 2 up to (but not including) entry point 4 in Figure 3-2].

(a) SITING attempts to allocate the unit set to its in-place site (which is also its goal site in this case).

(b) If the above cannot be done, the unit set is not allocated in this step.

(5) All remaining unallocated unit sets (those not allocated in step (4) are processed, in priority order, as follows [entry point 4 through end of chart in Figure 3-2]:

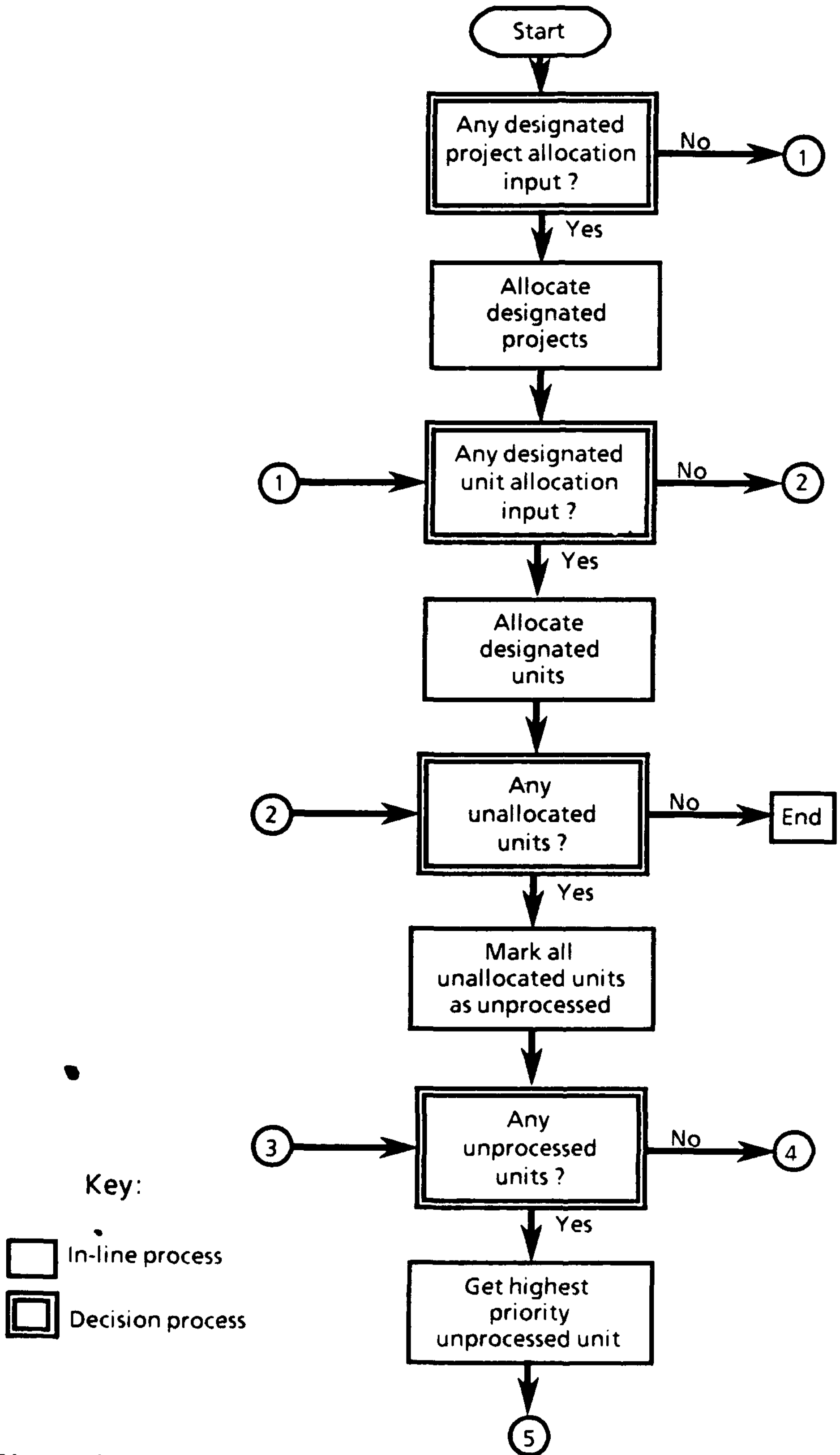

Figure 3-2. SITING Algorithm with Ignore Dispersion Option (Yrs 1-[Goal-1])
(page 1 of 3 pages)

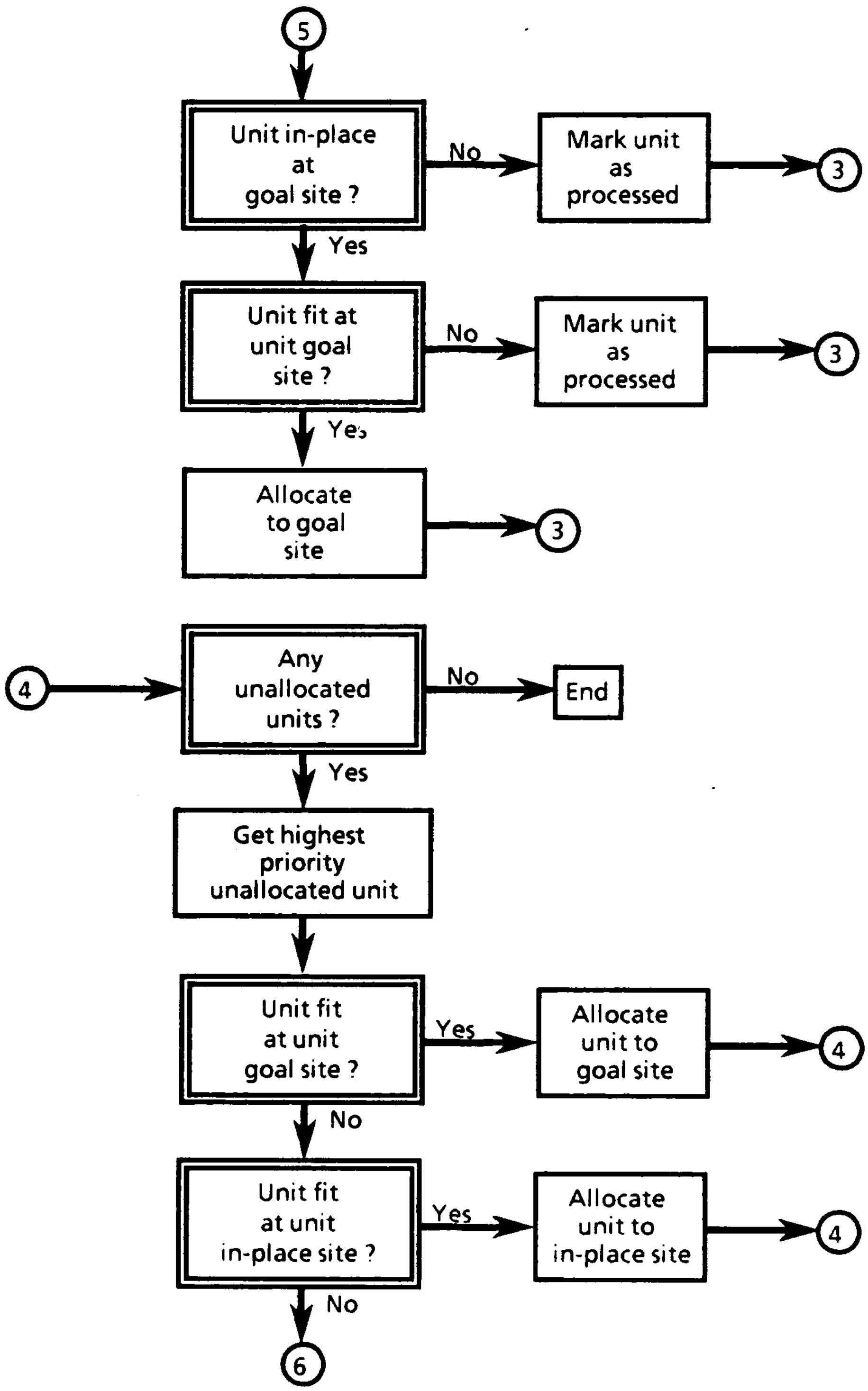

Figure 3-2. SITING Algorithm with Ignore Dispersion Option (Yrs 1-[Goal-1])
(page 2 of 3 pages)

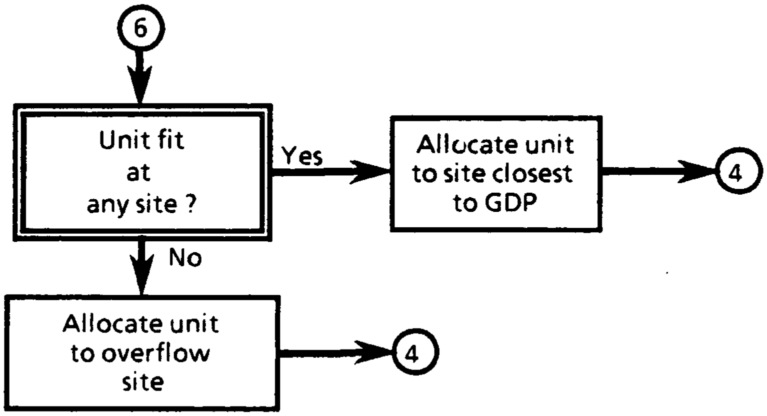

Figure 3-2. SITING Algorithm with Ignore Dispersion Option (Yrs 1-[Goal-1])
(page 3 of 3 pages)

(a) The unit set is allocated to its goal site, if possible.

(b) If the preceding is not possible, the unit set is allocated to its in-place site, if it can fit there.

(c) If that is not possible, the unit set is allocated to the site closest to its GDP that has space for it.

(d) If that is not possible, the unit set is allocated to an overflow site (site -99 or site -01).

(6) A numeric example (simplified) illustrating the processing for the ignore dispersion option in year 2 is shown in Appendix C (Example 3, page C-7).

3-8. ALGORITHM DESCRIPTION FOR "REDUCE DISPERSION" OPTION. As in the ignore dispersion option, SITING first processes (determines allocations) in the goal year (last year of timeframe). Next, it processes year 1 through year [goal-1] in that order. This option is more complicated than the previous one because both projects and unit sets must be considered. Some processing steps and terms are referenced by project name while others are referenced by unit set identity. Allocations for each year are done in two consecutive phases, denoted as the project allocation phase and the unit set allocation phase. As before, sections of logic flowchart are referenced in brackets below, after the subparagraph which describes processing in that section. The algorithm procedure is:

a. Processing of Goal Year. SITING initially processes the goal year of the timeframe in order to assign a unit set goal site for each unit set and a project goal site for each project. A unit set's unit set goal site is the storage site to which it is allocated by SITING in this processing of the goal year. Analogously, a project's project goal site is the storage site (if any) to which all units in that project are allocated by SITING during this processing of the goal year. A project with all of its units stored at the same site is said to be stored "intact" at that site. The unit set allocations to sites done by SITING in the goal year are the SITING solution siting plan for that year. As will be explained, these goal site assignments are also used in generating solution allocations for every other year. The reduce dispersion algorithm has both project allocation phases and unit set allocation phases. The goal year allocations of projects and unit sets are done solely on the basis of best unit set positioning (Objective 1) and reducing project dispersion (Objective 3). The algorithm logic in the goal year is shown in Figure 3-3. The sequence of processing is:

(1) If any designated project allocation input is present for this year, SITING attempts to allocate, in order of project priority, the input-specified projects to the input-specified sites [first three blocks of Figure 3-3].

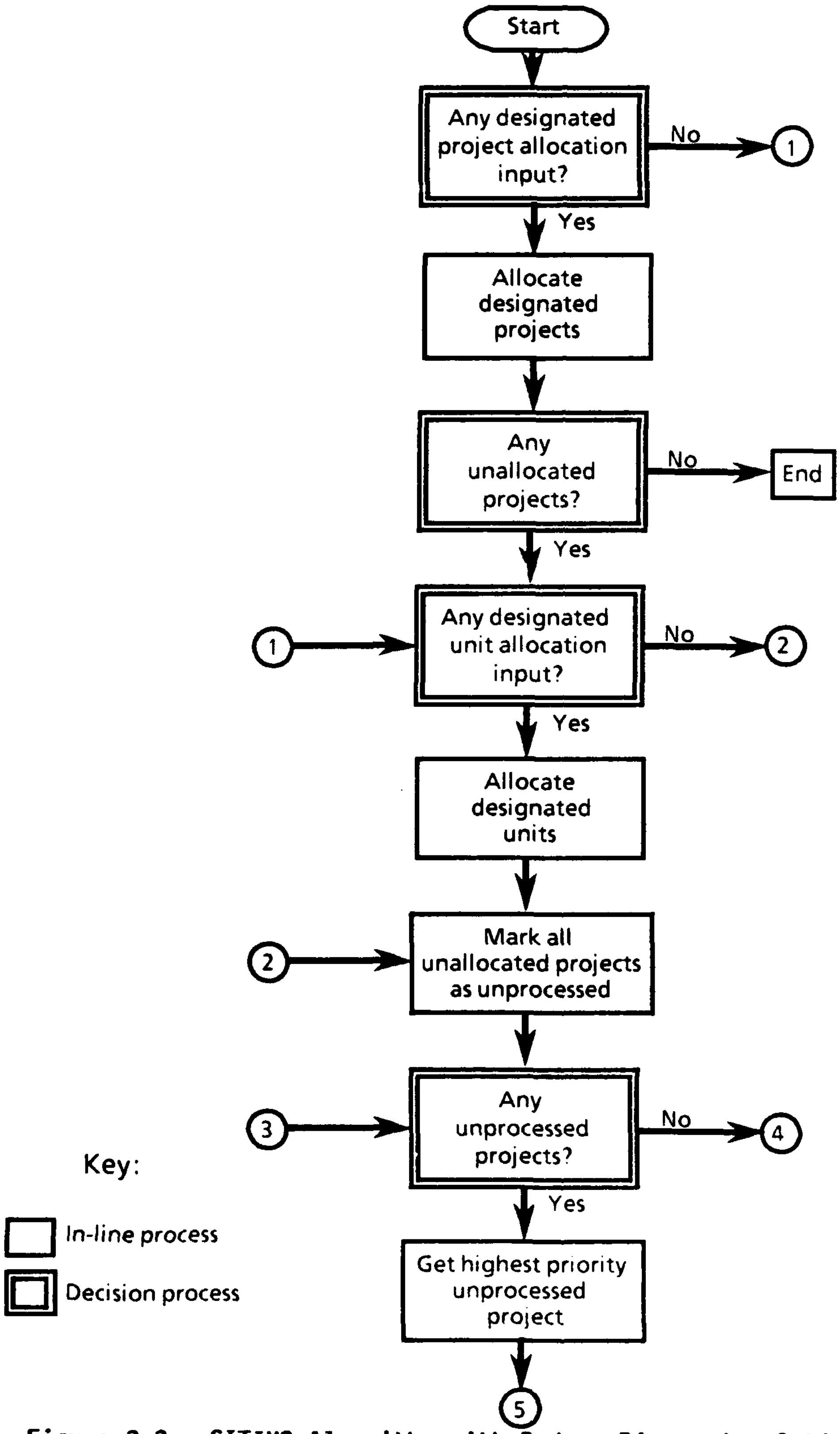

Figure 3-3. SITING Algorithm with Reduce Dispersion Option (goal year) (page 1 of 3 pages)

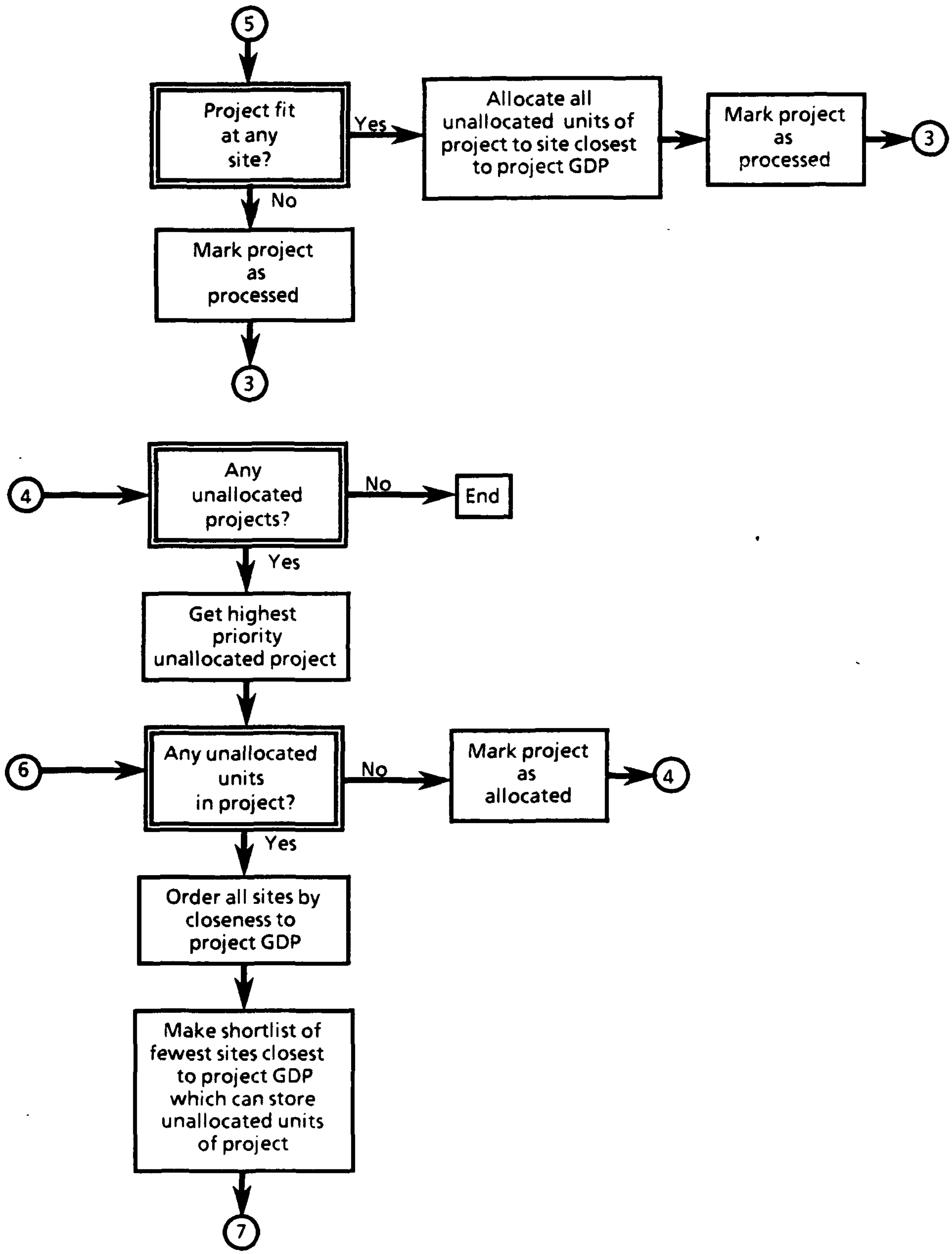

Figure 3-3. SITING Algorithm with Reduce Dispersion Option (goal year) (page 2 of 3 pages)

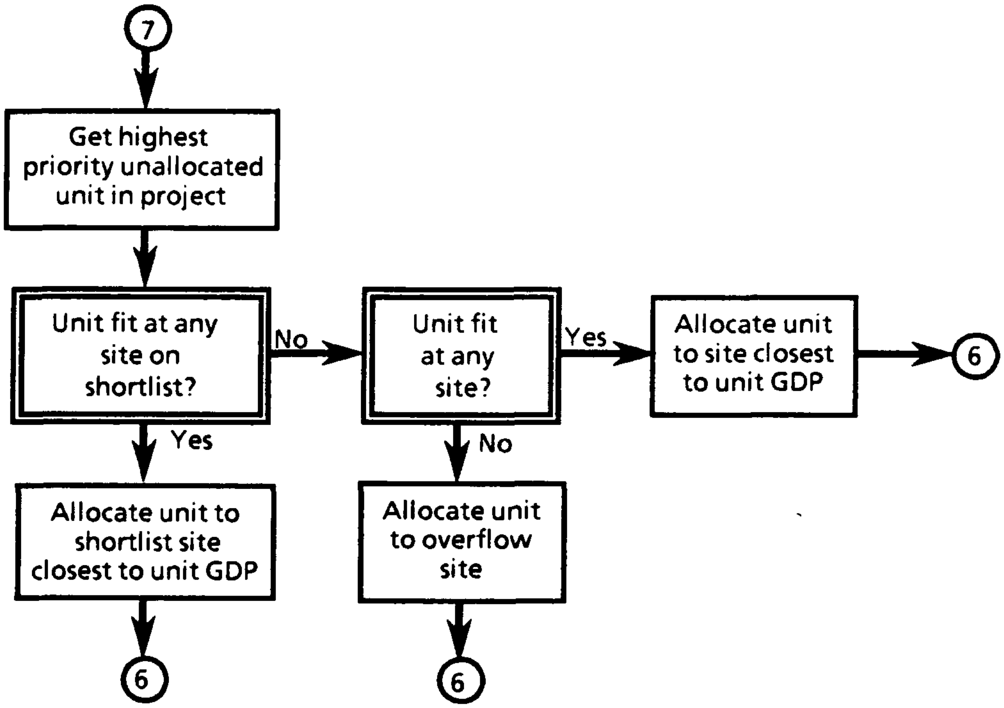

Figure 3-3. SITING Algorithm with Reduce Dispersion Option (goal year) (page 3 of 3 pages)

(2) After the designated project allocations are done, if any designated unit set allocation input is present for this year, SITING attempts to allocate, in order of input specification, the user-specified unit sets to the user-specified sites. If a unit set has already been allocated in step (1), it is not allocated in this step. [Entry point 1 up to (but not including) entry point 2 in Figure 3-3.]

(3) SITING constructs a site preference list for each project. To do this, a measure of average "closeness to project GDP" is computed for each project/site pairing in the scenario. Then, for each specified project P, all of the "closeness to project P GDP" measures, as computed for the specified project paired with each scenario site, must be rank-ordered by increasing ("closeness") value. This rank ordered list of site "closeness to project GDP" measures, and the associated sites, is defined as the site preference list for that specific project P. For a given project, P, and a given site, s, SITING computes the measure of "closeness to project GDP" for the project P/site pairing as follows:

(a) For each unit set, n, in the project, P, the "unit set n product" = [weight of unit set n] x [distance between GDP for unit set n and site s] is computed.

(b) All the "unit set n products" for all unit sets in project P are summed.

(c) The sum in (b) is divided by the total weight of all unit sets in project P. The result is the closeness to project P GDP measure for the project P/site s pairing.

> **NOTE:** The weight of a unit set is used so that heavy unit sets have a greater impact on the average than light ones. Since heavy units will usually require more resources to move than very light ones, they receive more consideration in closeness to project GDP calculations.

(4) The year is processed in a "project allocation phase" as follows. In increasing order of input project priority, SITING tries to allocate the entire unallocated portion of each project to the most preferred site on the site preference list for that project, i.e., the unallocated project portion is allocated intact, if possible, to the site with the smallest closeness to project GDP measure for that project. In this phase, either all unallocated unit sets in a project are allocated to the same site, or else no unit set of the project is allocated. If the unallocated project cannot be stored at any nonoverflow site, it is not allocated in this phase [entry point 2 through entry point 4 in Figure 3-3].

(5) After all projects have been processed in the project allocation phase, and after the designated unit set allocations are done, the remaining unallocated projects are processed in a "unit set allocation phase" as follows. In project priority order, each remaining unallocated project, P, is allocated as follows [entry point 4 through end of chart in Figure 3-3]:

(a) Starting with the site with the smallest closeness to project P GDP measure, and choosing sites according to increasing value of their closeness to project P GDP measure, the unoccupied space at each site is

accumulated until that accumulated total exceeds the total size (storage area) of all the unallocated unit sets of project P. All other sites are then excluded from consideration in the solution for this case. Denote the sites used to form the final accumulation as the "site shortlist" for project P. This shortlist is a smallest group of sites which probably has space available for the units of project P and which consists of preferred sites relative to positioning the unallocated unit sets in the project near their unit set GDPs.

(b) In increasing order of input unit set priority, each unit set of project P is allocated to the most preferred site on its unit set site preference list that is also on the site shortlist for project P; i.e., each unit set of the project is allocated to the site on the project P site shortlist that is closest to the unit set GDP. If no site on the shortlist has sufficient unoccupied space to store the unit set, it is allocated to the closest (to GDP) site on the full site preference list with space for the unit set. If no site on that list has room, the unit set is allocated to the notional overflow site -99 or to notional overflow site -01.

(c) The sites to which intact projects are allocated in the goal year will become project goal sites for projects in other years. The sites to which individual unit sets are allocated in the goal year will become unit set goal sites for unit sets in other years.

(6) A numeric example (simplified) illustrating the processing for the reduce dispersion option in the goal year is shown in Appendix C (Example 4, page C-9).

b. Allocation Processing for Other Years. The algorithm logic in the other years is shown in Figure 3-4. The in-place sites in year 1 are assumed to be the goal site allocations made previously. Units appearing for the first time in year 1 are assumed to have no in-place sites. (The input initial in-place siting is used only for assessment purposes.) The sequence of processing is:

(1) The SITING allocations of the previous year become the in-place site locations of the unit sets in the year being processed. Unit sets in year 1 and those appearing for the first time do not have an in-place site.

(2) The sites to which intact projects are allocated in the goal year will become goal sites for projects in other years. The sites to which individual unit sets are allocated in the goal year will become goal sites for unit sets in other years.

(3) As in the goal year, processing is in two consecutive phases, the project allocation phase and the unit set allocation phase.

(4) If any designated project allocation input is present for this year, SITING attempts to allocate, in order of project priority, the input-specified projects to the input-specified sites. [Start-up to (but not including) entry point 1 in Figure 3-4.]

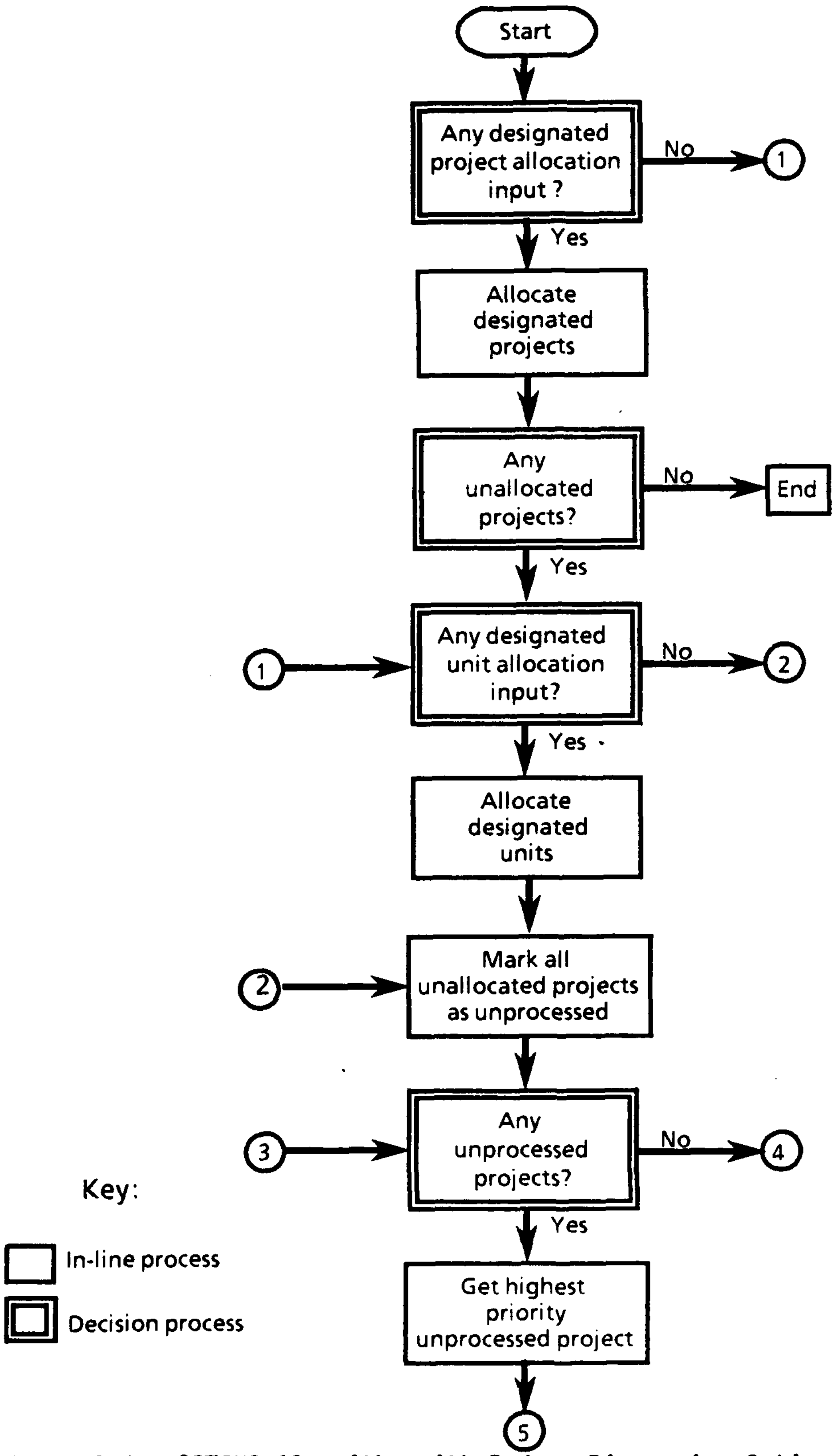

Figure 3-4. SITING Algorithm with Reduce Dispersion Option (Yrs 1-[Goal-1]) (page 1 of 5 pages)

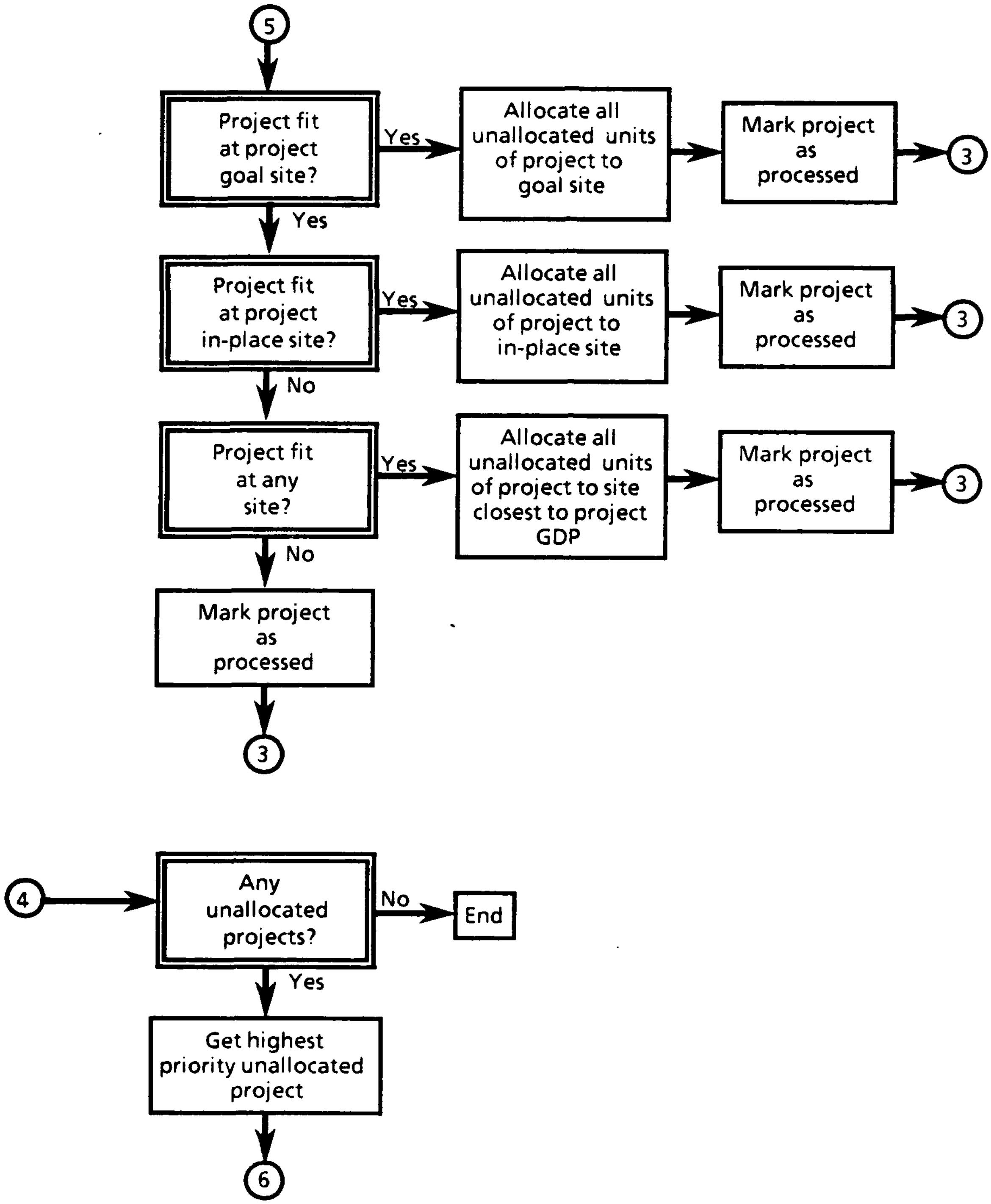

Figure 3-4. SITING Algorithm with Reduce Dispersion Option (Yrs 1-[Goal-1])
(page 2 of 5 pages)

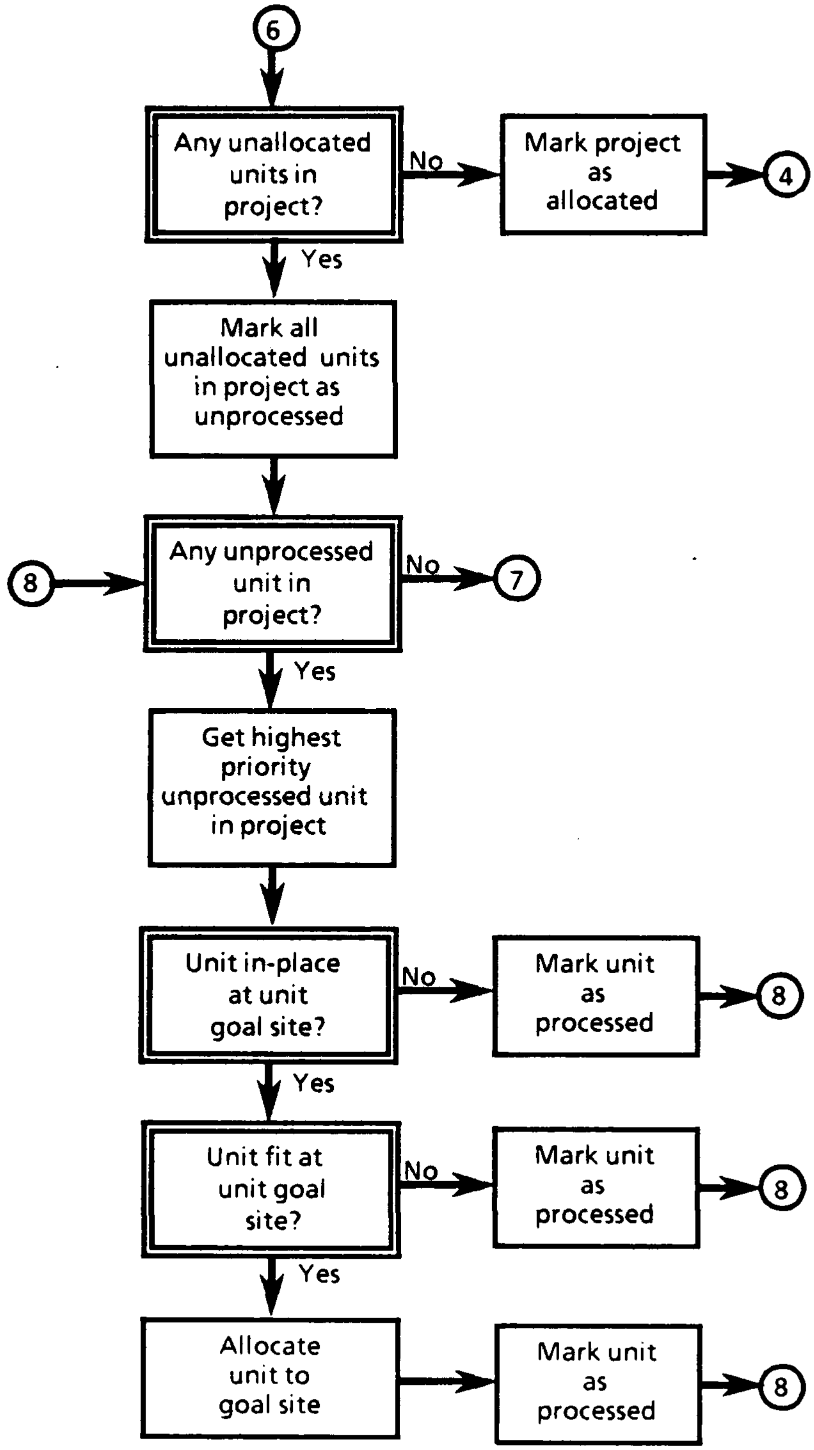

Figure 3-4. SITING Algorithm with Reduce Dispersion Option (Yrs 1-[Goal-1])
(page 3 of 5 pages)

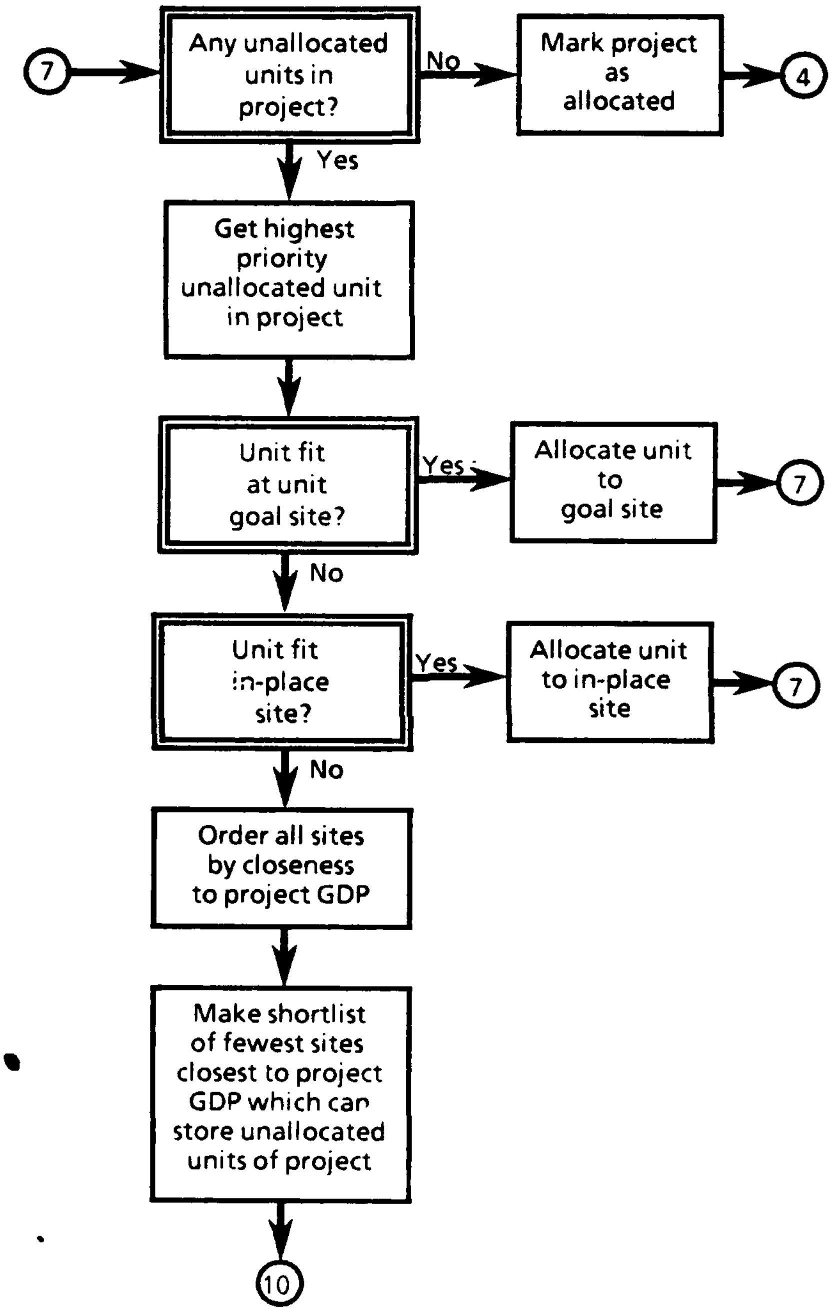

Figure 3-4. SITING Algorithm with Reduce Dispersion Option (Yrs 1-[Goal-1])
(page 4 of 5 pages)

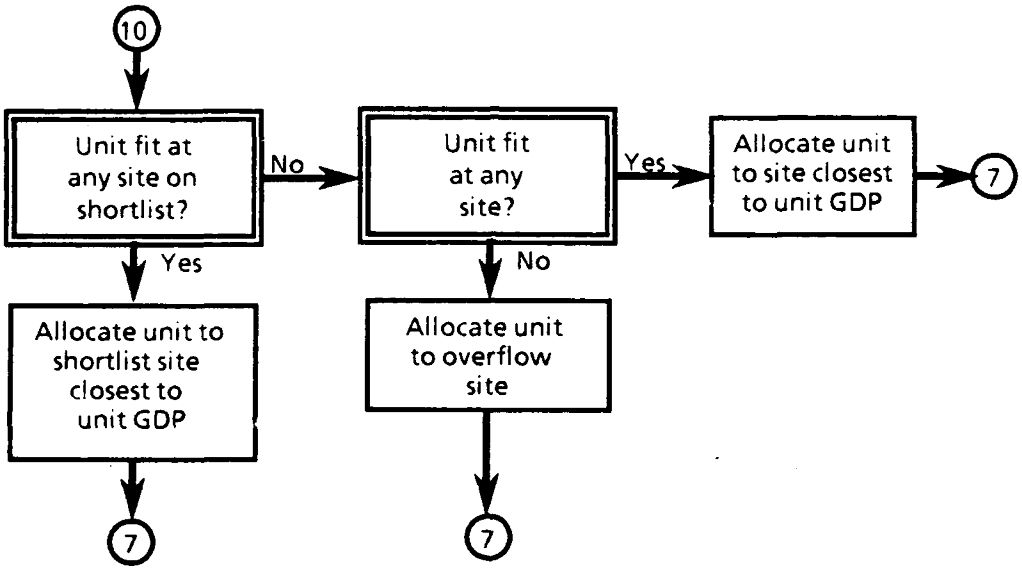

Figure 3-4. SITING Algorithm with Reduce Dispersion Option (Yrs 1-[Goal-1]) (page 5 of 5 pages)

(5) After the designated project allocations are done, if any designated unit set allocation input is present for this year, SITING attempts to allocate, in order of input specification, the user-specified unit sets to the user-specified sites. [Entry point 1 up to (but not including) entry point 2 in Figure 3-4.]

(6) In the project allocation phase, each unallocated project is processed in order of project priority as follows [entry point 2 through entry point 4 in Figure 3-4]:

(a) SITING tries to allocate the entire unallocated portion of each project to its project goal site.

(b) If that is not possible, SITING tries to allocate the entire unallocated project portion to its in-place site if it began the year intact at a single in-place site.

(c) If that is not possible, SITING tries to allocate the entire unallocated project portion to the most preferred available site on the site preference list for the project (i.e., the available site with the smallest "closeness to project GDP" measure for that project).

(d) If that is not possible, the project is not allocated in this phase, but the unit sets of the project are allocated in the unit set allocation phase.

(7) In the unit set allocation phase, the unallocated projects are processed in project priority order with their unit sets being processed in unit priority order. The following two phases are then performed. Each operates as follows on all unallocated unit sets of the project being processed [entry point 4 through end of chart in Figure 3-4]:

(a) All of the unit sets of the project that are in-place at their unit set goal site are processed first. Processing each unit set in order of unit set priority, SITING attempts to allocate it to its unit set goal site. If this cannot be done, it is not allocated in this step.

(b) The unit sets of the project that are not in place at their unit set goal sites and have not been allocated there are then processed. Processing in order of unit set priority, SITING attempts to allocate a unit set to its unit set goal site. If this is not possible, SITING tries to allocate it to its in-place site. If this cannot be done, the site shortlist of preferred sites that will contain the unallocated unit sets of the project is constructed in a manner exactly analogous to the way described in the goal year, and the unit set is allocated to the most preferred site on the project's site shortlist having space for it. If this is not possible, the unit set is allocated to the most preferred site on its site preference list. If this is not possible, it is allocated to the overflow site -99 or to overflow site -01.

(8) A numeric example (simplified) illustrating the processing for the reduce dispersion option in year 1 is shown in Appendix C (Example 5, page C-12).

CHAPTER 4

SITING INPUTS

4-1. INPUT REQUIREMENTS. The user specifies the SITING solution objectives to be addressed by entering scenario option inputs in response to on-screen menu prompts generated by SITING. In addition, SITING requires construction of four formatted ASCII data files. There are also two optional input files which can provide the user with added flexibility in problem definition and with additional output useful for solution analysis. The uses and capabilities of these optional inputs are described in Appendix C.

4-2. MENU-DRIVEN SCENARIO OPTION INPUT. The user selects SITING scenario options via keyboard-entered commands in response to an on-screen menu. First, the user is prompted on allocation scenario constraints, i.e., he decides whether he wishes allocations with site storage capacities as specified by the formatted input or whether he instead wishes unlimited storage capacity at all sites. Next, the user is prompted on treatment of project dispersion (storage of a project at multiple sites). The user can elect to keep project dispersion small, thereby treating all three objectives. Alternatively, he can elect to ignore project dispersion, thereby treating only Objectives 1 and 2 (see page 3-1). Elimination of Objective 3 enables further improvement (relative to the solution with Objective 3) of unit set positioning at the cost of increased unit set dispersal within projects. Lastly, the user is prompted whether the input should be altered to reduce the number of storage sites to a minimum. Although SITING does not seek the fewest number of sites over which to store all of the unit sets, this option does enable the user to have SITING input data altered to eliminate certain sites that may not be needed. The allowable combinations of these options are summarized in Table 4-1. These user-specified options are described below.

Table 4-1. Allowable Combinations of Menu-driven SITING Input Options

Allocation scenario	Project dispersion	Limit on redundant sites
1. Available storage case	Ignore dispersion	Allowed
2. Available storage case	Reduce dispersion	Allowed
3. Unconstrained storage case	Ignore dispersion	NA
4. Unconstrained storage case	Reduce dispersion	NA
5. Baseline case	NA	NA

NOTE: NA - not applicable.

a. Allocation Scenario Options. The onscreen choices presented to the user are:

(1) Available Storage Case. Each storage site's capacity is as specified by the formatted input. SITING will determine siting plan allocations.

(2) Unconstrained Storage Case. All storage sites have essentially unlimited storage capacity. The resulting SITING siting plan will then allocate every unit set to its best positioned storage site. This is a benchmark "best case."

(3) Baseline Case. No siting plan allocations are to be determined. Instead, the "goodness" of the input in-place siting, using unit set characteristics for year 1, is assessed in light of how closely SITING Objectives 1, 2, and 3 are met. The output described in Chapter 5 and Appendix E is produced for the current year (represented by the input in-place siting) by simple arithmetic. The SITING allocation algorithm is not exercised in this case.

The option (1) above produces the standard SITING solution siting plan. Option (2) produces a best case siting plan because storage constraints are removed. Option (3) above gives MOEs for a first year solution siting plan based on doing nothing and just letting unit sets stay at the current in-place sites. Comparison of these MOEs with those for options (1) and (2) gives insights into how the standard SITING solution compares to doing nothing and to a best case (albeit an unachievable one).

b. Project Dispersion Options. The onscreen menu choices are:

(1) Ignore Dispersion Option. If this option is specified, then only Objectives 1 and 2 will be sought. Objective 3 will be ignored. In this case, unit sets will be stored close to unit GDPs, and resiting turbulence in each year is kept small. If this option is not specified, then the user must specify the reduce dispersion option.

(2) Reduce Dispersion Option. If this option is specified, then Objectives 1, 2, and 3 will be sought. In this case, unit sets will be stored close to unit set GDPs, resiting turbulence in each year is kept small, and the stored unit sets of each individual project will be dispersed over as few sites as possible while satisfying Objectives 1 and 2. If this option is not specified, then the user must specify the ignore dispersion option.

c. Redundant Site Limit Option. This option need be set only if the user has previously chosen the available storage case option for the allocation scenario. The user is asked (via the screen) how many redundant sites will be allowed. If all sites are to be available for use, the user should enter a value larger than the number of sites in the problem. The onscreen menu suggests a value of 99 as a response in this case. Otherwise, if the user wants only the largest capacity sites to be considered for storage of the solution, he also has the option of allowing SITING to determine a solution for a minimum (as determined by SITING) number of sites. Prior to "solving" for unit set allocations in each year, SITING estimates the minimum number of input storage sites which are needed to store all of the unit sets that must be allocated (stored) in that year. This minimum number of sites then forms a basic "sites allowed" list for the problem that year. The associated sites are those with the largest storage capacity. SITING then internally marks

all of the other sites as "redundant sites." The user must specify the maximum number of the redundant sites which they will allow to be added to the basic sites allowed list. If the user specifies "n" of the redundant sites to be allowed, SITING will take the n largest site capacities of the redundant sites and will transfer the associated sites from the redundant sites list to the allowed sites list. The remaining redundant sites are then removed from the problem for that year. The method used by SITING to determine the basic lists of allowed sites and redundant sites in each year is as follows:

(1) SITING internally rank-orders all sites, in that year, by decreasing site capacity.

(2) Proceeding down the ranked site list, SITING then accumulates the sum of site capacities, beginning with the largest site, and marks the point on the list at which accumulated capacity exceeds the sum of all unit set areas, over all unit sets to be allocated, in that year. The site at this point on the list, and all sites ranked above it (i.e., with larger capacity) comprise the basic allowed sites list. All sites ranked lower (i.e., with less site capacity than the marked site) on the list are designated as redundant sites and comprise the redundant sites list.

(3) If the user enters a number, n, in response to the interactive menu prompt, then the first n redundant sites in the ranked site list are transferred to the allowed sites list and are removed from the redundant sites list.

(4) All of the sites remaining in the redundant sites list are eliminated from consideration for allocations. (This is done by setting their site capacities to zero.)

(5) All of the sites in the allowed sites list are unaltered in their input-specified characteristics and comprise the entire set of sites allowed to receive unit set allocations

This option alters the SITING input data, but the SITING algorithm procedure is not affected. For each year of the timeframe, this option selects the fewest possible sites which, together, will accommodate all unit sets and restricts SITING algorithm consideration to those selected sites. The option effectively eliminates the other "unneeded" sites in the scenario for that year. The resulting "fewest possible sites" solution will not meet Objective 1 (closeness to unit GDP) as well as a solution allowing allocations over more sites. The number of sites eliminated in the input alteration is indicative of the magnitude of the tradeoff.

4-3. PREPARATION OF FORMATTED INPUT FILES. This paragraph treats SITING inputs which are prepared before run time by the user (or his agent). The input files described in this paragraph must be resident in the PC directory containing the executable SITING program. This paragraph defines the format and content of the required SITING input files. Preparation of SITING input requires the creation of four separate ASCII files which must be named File8.txt, File9.txt, File10.txt, and File11.txt. The structure of these input files is given below. The input files must be named exactly as described below. Table 4-1 summarizes the required POMCUSITE SITING input

files. The use and formats of optional input files which may provide added flexibility are given in Appendix E. The SITING default is an 8-year timeframe. If fewer than 8 years are run, an optional input file, denoted as input File15.txt must be created and used (see Appendix E). Optional input File15.txt is also reqired if the user wishes SITING to assess the generated solution siting plan relative to the problem objectives.

Table 4-2. Required SITING Input Files

File name	Descriptive name
File8.txt	Unit definition file
File9.txt	Site definition file
File10.txt	Project definition file
File11.txt	In-place definition file

a. Unit Definition File. The unit definition file must be named File8.txt. The ID for each unit set must be a unique integer between between 1 and 2,000. (At most, 1,000 of these integers can be used in any one problem.) Over the entire timeframe, no more than 1,000 unit sets are allowed to be defined. These unit set IDs are the only "names" which SITING uses to identify a unit set. For each year, File8.txt gives each unit set ID number (i.e., the name of the unit within the model), the geographic location of the GDP for that unit set, the unit set size (i.e., the storage area it occupies), and the unit set priority. The structure of this file consists of as many consecutive data blocks as there are years, one for each year processed. The blocks must be in order of the years processed, i.e., the first block must be for the first year, the second block for the second year, etc. Each block must have the following structure. All integer input must not contain a decimal and must be right justified in the indicated data field.

(1) Record 1

Columns 1-3: the integer numeric ID of the year being processed in this block. These numeric IDs must be between 1 and 8, with 1 representing the first year, 2 representing the second year, etc.

(2) Following record 1 are records describing the unit sets of this block. These consist of exactly one record for each unit set being processed in the year indicated by record 1 of the block. These records (for the unit sets in the block) need not be in any specific order. Each record has the following structure:

Columns 1-5: the positive integer numeric ID of this unit set (the unit set described in this record). This numeric ID must be between 1 and 2,000. Other values are not allowed. A specified unit set must have the same numeric ID over all the years (blocks) in which it is defined in this file.

Columns 6-15: the integer floor space area (square meters) required for the sum total of all equipment in this unit set.

Columns 16-25: the integer total weight (short tons (STON)) of all equipment in this unit set.

Columns 26-30: the integer in the latitude location of the unit set GDP associated with this unit set.

Columns 31-32: the integer minutes in the latitude location of the unit set GDP associated with this unit set.

Columns 33-34: the integer seconds in the latitude location of the unit set GDP associated with this unit set.

Column 35: the character N is entered to denote north latitude for the GDP location. (This SITING configuration does not process south latitudes.) The Great Circle distance calculations must be done for locations in the same hemisphere. The program code has been written for the northern hemisphere only.

Columns 36-40: the integer degrees in the longitude location of the unit set GDP associated with this unit set.

Columns 41-42: the integer minutes in the longitude location of the unit set GDP associated with this unit set.

Columns 43-44: the integer seconds in the longitude location of the unit set GDP associated with this unit set.

Columns 45: a character E or a character W is entered to denote the type of longitude for the GDP location. An E is entered for east longitude. A character W is entered for west longitude. Both east and west longitudes are allowed.

Columns 46-52: the integer numeric priority of this unit set. This should be a positive number. The lower the value of this numeric priority, the higher is the actual preference priority. Thus, a numeric priority 1 is "top priority." Allocations of unit sets within a project will be done according to the (increasing) order of unit set priority. A higher unit set priority increases the likelihood of the associated unit set being allocated to a site positioned close to that unit set's GDP. **NOTE:** the SITING user must not set too large a value for input unit set priority; otherwise, the calculations for combined unit set priority will cause numeric overflow. To ensure nonoverflow, the user should limit values for unit set priorities to numbers less than 1,000,000 while values of project priorities are constrained to values less than 100.

(3) Final record of year block:

Columns 1-5: the integer -9 is entered here to signal the end of the year block to the model.

b. Site Definition File. The site definition file must be named File9.txt. For each storage site in each year, it gives the storage capacity (in floor area) and the location of each storage site. Each site must appear in all years. If it is not present in a year, it must be designated as having a zero capacity. Over the entire timeframe, a maximum of 28 different sites are allowed to be defined. The structure of this file consists of as many consecutive data blocks as there are years, one for each year processed. The blocks must be in order of the years processed, i.e., the first block must be for the first year, the second block for the second year, etc. Each year block must have the following structure. All integer input must not contain a decimal and must be right justified in the indicated data field.

(1) Record 1

Columns 1-3: the integer numeric ID of the year being processed in this block. These numeric IDs must be between 1 and 8, with 1 representing the first year.

(2) Following record 1 are records describing the storage sites of this block. These consist of exactly one record for each storage site being processed in the year indicated by record 1 of the block. Each record has the following structure:

Columns 1-5: the integer numeric ID of this storage site (the storage site described in this record). A specified site must have the same numeric ID over all the years (blocks) in which it appears.

Columns 6-15: the integer storage capacity, in terms of maximum storage area (square meters) available at this storage site.

Columns 16-20: the integer degrees in the latitude location of this site.

Columns 21-22: the integer minutes in the latitude location of this site.

Columns 23-24: the integer seconds in the latitude location of this site.

Columns 25: the character N is entered to denote north latitude for the site location. (This SITING configuration does not process south latitudes.)

Columns 26-30: the integer degrees in the longitude location of this site.

Columns 31-32: the integer minutes in the longitude location of this site.

Columns 33-34: the integer seconds in the longitude location of this site.

Column 35: a character E or a character W is entered to denote the type of longitude for the site location. An E is entered for east longitude, a character W is entered for west longitude.

(3) Final record of year block:

Columns 1-5: the integer 999 is entered here to signal the end of the year block to the model.

c. Project Definition File. The project definition file must be named File10.txt. For each project in each year, it defines the priority assigned to each project. It also defines the unit sets (according to the numeric unit set ID defined in File8.txt) which comprise each project. A maximum of 25 projects is allowed to be defined in a year. The structure of this file consists of as many consecutive data blocks as there are years, one for each year processed. The blocks must be in order of the years processed, i.e., the first block must be for the first year, the second block for the second year, etc. Each year block must have the following structure. All integer input must not contain a decimal and must be right justified in the indicated data field.

(1) Record 1

Columns 1-3: the integer numeric ID of the year being processed in this block. These numeric IDs must be between 1 and 8, with 1 representing the first year.

(2) Following record 1 are sub-blocks. Each sub-block corresponds to a project to be processed in this year (block). There must be exactly one sub-block for each project processed. The sub-block structure for each project is as follows:

(a) First Record. This identifies the project for this sub-block.

Columns 1-5: the positive integer numeric ID of the project being processed in this block. This ID must have a value between 1 and 30. Other values are not allowed. A specified project must have the same numeric ID in all years in which it appears.

Columns 6-10: the positive integer numeric priority for this project. A low value denotes high priority. Thus, a priority of 1 is "top priority." SITING will combine the project priority with the unit set priority (from File8.txt) so that unit sets will be allocated in (increasing) order of their associated project's priority. Unit sets within a project will be allocated in (increasing) order of unit set priority (as defined in File8.txt). **NOTE:** the SITING user must not set too large a value for input project priority; otherwise, the calculations for combined unit set priority will cause numeric overflow. To ensure nonoverflow, the user should limit values for project priorities to numbers less than 100 while values of unit set priorities are constrained to values less than 1,000,000.

(b) Other Records. The next set of records in the sub-block for this project identifies the unit set IDs (as defined in File8.txt) of the unit sets comprising this project. There is exactly one record here for each unit

set in this project. These unit set records do not need to be entered in any particular order. The structure of a record for a specified unit set in this project is:

Columns 1-5: the positive integer numeric unit set ID, as defined in File8.txt, for this unit set. This ID must cross-reference with File8.txt.

(c) Final Record of Project Sub-block

Columns 1-5: the integer -9 is entered here to signal the end of the project sub-block to the model.

(3) Final Record of Year Block. After the last sub-block, the positive integer 9999 is entered to signal the end of the block for that year.

d. In-place Definition File. The in-place definition file must be named File11.txt. For only the start of the first year, it gives the initial (in-place) storage site location of all unit sets defined in the Unit Definition File. If not all unit sets in File8.txt are accounted for here, the unit sets without a specified in-place location are treated as being at a notional site (denoted as site -01 in SITING). File11.txt is not used by SITING to determine allocations. It is only used to assess certain MOEs on resiting turbulence for the SITING solution. The model will allocate correctly even if File11.txt is absent. (In such a case, the resiting MOEs assessed for the first year will be invalid). If File11.txt is present, it must be consistent with File8.txt and File9.txt. The structure of this file consists of a series of site data blocks. One block must appear for each site which contains in-place unit sets. These blocks do not have to be in any particular order. There should be no site block for a site with no in-place unit sets. Each site block must have the following structure. All integer input must not contain a decimal and must be right justified in the indicated data field:

(1) Record 1

Columns 1-5: the integer numeric site ID, from File9.txt, of the site being processed in this site block. These numeric IDs must cross-reference with the sites defined in File9.txt.

(2) Following record 1 are records stating the numeric unit set IDs (from File8.txt) of all the unit sets stored in place (at start of initial year) at the site being processed in this block. These consist of exactly one record for each unit set in place at the indicated site. Each record of the site block has the following structure:

Columns 1-5: the integer numeric site ID, from File8.txt, of the specified unit set which is in place at this site. These numeric IDs must cross-reference with the unit sets defined in File8.txt.

(3) Final record of site block: after the last unit set record for this site, the integer -9 is entered to signal the end of the block for that year.

The above-constructed files must be copied into files with the associated names (File8.txt, File9.txt, File10.txt, and File11.txt) and must be resident in the same file directory (on the PC) as the SITING executable program. If optional input file15.txt (see Appendix E) is omitted, SITING will assume that the timeframe has exactly 8 years.

CHAPTER 5

SITING OUTPUTS

5-1. SITING OUTPUT FILES. SITING generates a standard set of output files if optional inputs (described in Appendix C) are not used. Table 5-1 summarizes the standard SITING output files which describe the SITING solution siting plan. The structure of each standard output data file is described in this paragraph. These will always be generated with the names File12.txt, File16.txt, File19.txt, and File21.txt. SITING also produces as output a text output File19.txt. It has no fixed format. File19.txt records the scenario problem conditions chosen by the user in the interactive menu input. No format description is given below for output File19.txt because it has none. Optional extra file output which is available through use of optional input is described in Appendix C. These optional outputs include MOEs assessing how well the solution meets the problem objectives.

Table 5-1. Basic SITING Output Files

File name	Descriptive name
File12.txt	Siting allocations listing
File16.txt	Site-site distance listing
File21.txt	Unit set site preference lists
File19.txt	Scenario conditions list

5-2. SITING ALLOCATIONS LISTING FILE OUTPUT. Output File12.txt is the siting allocations listing. For each year processed, it shows all unit set allocations for every unit set present in either that year, in any year preceding that year, or in the goal year. As a consequence, in each year group, this file may show allocations of unit sets which are actually not present that year (e.g., a unit set may be present in the goal year, but not in the year being processed). When "not present" in a year being processed, a unit set in this file is treated as having a size (storage area) of zero for allocation purposes in SITING. When not present in the year being processed, a unit set will always be entered in this file as having a unit priority (columns 15-21) of 0 and a "distance to unit GDP" (columns 11-14) of -1. (This marks these units so that the user may filter them out with a postprocessor if desired.) Within each year, all allocations for that year are sorted (grouped) by allocation site. The items, in order of column position, are:

a. Columns 1-3: the integer year being processed. The first year is 1, the last is 8 (for an 8-year case).

b. Columns 4-7: the integer numeric unit set ID of the unit allocated.

c. Columns 7-10: the integer numeric site ID (descriptor) of the storage site to which this unit set is allocated in this year. This is the allocation site.

d. Columns 11-14: the integer distance (in km) of the allocation site from the unit set GDP for the allocated unit set. If this value is -1, the associated unit set is not really present in this year.

e. Columns 15-21: the integer unit set priority (from input File8.txt) of the allocated unit set. If this value is 0, the associated unit set is not really present in this year.

f. Columns 22-24: the integer numeric site ID of the storage site which is the closest to the allocated unit set's GDP.

g. Columns 25-27: the integer numeric site ID of the storage site which is the in-place site (previous year's allocation site) for the allocated unit set.

5-3. SITE-SITE DISTANCE LIST FILE OUTPUT. Output File16.txt is the site-site distance listing created whenever SITING executes. It gives distances, in kilometers, between all paired storage sites in each year processed. The items in the file are:

a. Columns 1-3: the integer numeric site ID of first member of site pair.

b. Columns 4-6: the integer numeric site ID of second member of site pair.

c. Columns 7-13: a decimal number equal to the distance, in km, between members of this site pair.

5-4. UNIT SET PREFERENCE LIST FILE OUTPUT. File21.txt consists of the unit set site preference lists. For each unit set processed in a year, this file gives the rank ordering of all storage sites relative to the distance from that storage site to the GDP for that unit set. The associated distance is also given. Such a site preference list is very useful because, given a SITING allocation of a unit set to a specific site, a look at the site preference list for that unit set will indicate how close to optimum this allocation is. Since the top item on the list is the site closest to the unit set GDP, that "top site" is the optimum site. This file consists of a series of such site preference lists for all unit sets processed in the final year (of the timeframe). The order of the lists is the order of the unit sets in the input File8.txt. Each record in File21.txt gives site preference information for a specific combination of year, unit set, and site. The structure of each record is:

a. Columns 1-6: the integer numeric unit set ID for this record

b. Columns 7-11: the integer numeric site ID for this record

c. Columns 12 -21: a decimal number equal to the distance (in km) between the indicated unit set's GDP and the indicated site.

d. Columns 22-26: the year for which this record is applicable (1 through 9, where 9 is equivalent to the goal year).

CHAPTER 6

SITING OPERATION

6-1. PURPOSE. This chapter describes the procedure for operation of SITING on a PC, given that the input files are prepared and in place. It consists of a "walkthrough" of the onscreen prompts which will be generated by SITING to solicit scenario option input from the user. The time required for processing depends on the scenario. Test cases with 500 unit sets and 24 sites required approximately 20 minutes of clock time to process the full (8-year) timeframe.

6-2. PRELIMINARY PREPARATION

a. The PC must have 450K of RAM available for SITING program execution.

b. The SITING program must be resident in a directory on the PC hard disc drive or on a Bernoulli disc drive.

c. The SITING input files must be resident in the same hard drive PC directory as the SITING program. These files must be correctly constructed in accordance with the input specifications in Chapter 3.

d. Sufficient additional space must be available in the disc directory (containing SITING program and input) to contain the SITING output files. (Approximately 500K-1,500K may be needed.) This space is overwritten each time SITING is executed.

6-3. PROCEDURE FOR SITING OPERATION. This paragraph describes, via a "walk-through" example, all of the user procedures needed to generating a siting plan solution while executing SITING. This walkthrough assumes that the user's working directory contains the SITING executable program which is named SITING.EXE. The example treats a case with a 8-year timeframe in which only standard output is produced.

a. Step 1 - at the DOS prompt, enter (the command) SITING. The following prompt will occur shortly:

```
    ** 1.  LOAD DATA FOR THIS CASE INTO DRIVE **
    ** 2.  AFTER DATA IS LOADED, PUSH ENTER KEY **

Pause - Please enter a blank line (to continue) or DOS command.
```

b. Step 2 - as indicated by the prompt, ensure that the input data for this case are resident in the directory containing the SITING program. Then push the ENTER key on the PC. The following prompt then appears.

```
** TO ABORT THIS RUN AT ANY TIME, PUSH [CTRL-C] **

CHOOSE ONE OPTION

 1. AVAILABLE STORAGE CASE (ENTRY = 1)

 2. UNCONSTRAINED STORAGE CASE (ENTRY = 2)

 3. BASELINE CASE (ENTRY = 3)
```

c. Step 3 - the user should be able to abort the run at any time by simultaneously depressing the [CTRL] key and the C key on the PC. To generate a siting plan in this example, a choice must be made between options 1 and 2, since option 3 performs assessment only and does not generate any solution siting plan (see paragraph 4-2a, Chapter 4). Assume that the user wishes the plan to be based on the input storage site constraints. A value of 1 is then entered here. The default (if the user cannot properly enter an option) is the available storage case. The following prompt then appears. It confirms the choice made and allows the user to continue defining input or to skip further input definition, in which case defaults will be set for all inputs not defined thus far. In this and in all other SITING-generated menu prompts, SITING sets the default response if the user fails for five consecutive times to give an allowable value in response to the prompt. At the end of the input definition phase, the user is allowed, via a prompt, to change any input settings he/she has made thus far. The prompt shown below appears whenever this input is set, regardless of whether it is a first definition or a redefinition. If it is a redefinition, the user will likely want to skip repeating his/her past definitions of succeeding inputs.

This confirms the choice made and allows the user to continue defining input or to skip further input definition, in which case defaults will be set for all inputs not defined thus far.

d. Step 4 - in this case, it is a first definition, so enter a zero. Since a choice was made to generate siting plan allocations, the following prompt then appears:

```
1. REDUCE PROJECT DISPERSION OVER SITES (ENTRY = 1)

2. IGNORE PROJECT DISPERSION (ENTRY = 2)
```

If a user wants to treat all three SITING problem objectives (as described in Chapter 1 of this paper) a value of 1 is entered. If a user wants to ignore Objective 3 and emphasize Objectives 1 and 2, a value of 2 is entered. This prompt does not appear when the baseline case option is specified because no allocations are done in that case.

e. Step 5 - assume a choice to treat all three objectives. Enter a 1. The following prompt then appears:

```
REDUCE PROJECT DISPERSION (CAN REDEFINE LATER).

ENTRY = 0 TO CONTINUE

ENTRY = 9 TO SKIP REMAINING INPUT OPTIONS
```

This, as for the first input, confirms our choice and, if this is a redefinition, allows us to skip remaining input settings.

f. Step 6 - since this is a first definition, enter 0 to continue. Since the available storage case was specified, the following prompt next appears:

```
HOW MANY REDUNDANT SITES (BEYOND THE ESTIMATED MINIMUM
NEEDED EACH YEAR TO STORE ALL UNIT SETS) DO YOU ALLOW?
(ENTRY = NUMBER ALLOWED)
(ENTRY = 99 ALLOWS ALL SITES)
```

SITING will internally rank order all sites, in each year, by decreasing site capacity. It will then accumulate the sum of capacities through the ranked site list, beginning with the largest site, and will mark the point on the list at which accumulated capacity exceeds the sum of all unit set areas in the year. All sites ranked lower (i.e., with less site capacity than the marked site on the list are designated redundant. If the user enters a number, n, in response to this prompt, then only the first n redundant sites in the ranked site list are made available for allocations. A redundant site is treated by the algorithm as if the site capacity is zero. The default is 99 (i.e., make all sites available). In fact, any number equal to or greater than the number of redundant sites will make all sites available. This prompt does not appear when the unconstrained storage case is chosen because, by definition, all sites must be available for that case (otherwise, storage would be constrained). This prompt does not appear when the baseline case option is specified because no allocations are done in that case.

g. Step 7 - assume a choice to make all sites available at their capacity as specified in input file9.txt. A value of 99 is then entered. The following prompt then appears:

```
NR ESTIMATED REDUNDANT SITES ALLOWED = 99

ENTRY = 0 TO CONTINUE

ENTRY = 9 TO SKIP REMAINING INPUT OPTIONS
```

h. Step 8 - this is a redundant prompt confirming the input setting. Enter a 0. The following prompt then appears:

```
FOLLOWING ARE ITEM SETTINGS:

 (ITEM NR = 1): THIS IS AVAILABLE STORAGE CASE
 (ITEM NR = 2): REDUCE PROJECT DISPERSION
 (ITEM NR = 3): NR OF EST. REDUNDANT SITES ALLOWED
= 99

 SET ENTRY = ITEM NR TO REDEFINE ASSOCIATED INPUT
 SET ENTRY = 0 TO BEGIN RUN
```

This allows the user to review all item option settings and to change any before the SITING allocation algorithm begins. For example, to change item 2 option setting, one simply enters a 2 and repeats the definition process described in Step 5 above. In such a case, after the redefinition of item 2 option the user can "skip remaining input options" to avoid repeating the redefinition of item 3.

i. **Step 9** - in this case the SITING run is initiated by entering a 0. Early in the SITING run, the following prompt appears:

```
** SITING ALLOCATIONS BEGIN **
```

If a fatal data error is found or if certain nonfatal data discrepancies are found, a warning is printed on the screen. A key to the meaning of these messages is given in Appendix B. The run will abort if the detected error is fatal. The run may also abort because of an undetected data error. As SITING execution proceeds, each year, from year 1 through the final year, is processed (i.e., siting allocations are determined) in turn. When the model completes processing for a year N, the following message is written to the screen:

```
** SITING ALLOCATIONS COMPLETED IN YEAR N
```

If the run proceeds normally, at the completion of SITING processing for all years, the following message appears on the screen and is followed by the DOS prompt:

```
** ALLOCATIONS COMPLETED FOR ALL YEARS **
```

At this time, the site-site distance listing output file, named File16.txt, is ready to be used by the MOVER module of POMCUSITE. In addition, an output file named File12.txt is ready to be used by MOVER and by SITING module report generators. SITING execution is now complete. Output files File12.txt, File16.txt, and File21.txt have been generated. If the user wishes to save a SITING output file permanently, that file must be copied to a user-specified permanent file, otherwise it will be overwritten the next time SITING executes on this PC. All files are in ASCII and are less than 80 characters in width, so they can be printed on standard size paper. The case described above did not use optional input file15.txt. In a case in which the user inputs optional input file15.txt, a solution assessment would

be processed immediately after the allocation phase described above. In such a case, additional output files, described in Appendix E, would also be produced.

6-4. USER-SPECIFIED SCENARIO OPTIONS

a. Available Storage versus Unconstrained Storage. The available storage option is the only one that yields a practical and realistic siting solution. However, the siting plan produced by the unconstrained storage option shows how the storage sites should be "sized" in order to store each and every unit set at its optimally positioned (relative to SITING problem Objective 1) storage site. Under the unconstrained storage option, SITING will tend to store unit sets significantly closer to their unit set GDPs than under the available storage option.

b. Ignore Dispersion versus Reduce Dispersion. With the ignore dispersion option set, the resulting solution siting plan will emphasize SITING problem Objective 1 (positioning close to unit set GDP), but will not treat objective 3 (reduction of project dispersion). That solution plan will likely have many projects stored over several sites. If the reduce dispersion option is used, the solution siting plan may greatly reduce the number of sites over which an average project is dispersed, but this solution will be worse (than the ignore dispersion solution) relative to storing unit sets close to their GDPs. Objective 1 is traded off against Objective 3. Only experimentation can determine the extent of this tradeoff.

c. Number of Redundant Sites Allowed. The most practical thing is to operate with 99 redundant sites allowed so that all sites are considered. If few (or no) redundant sites are allowed, SITING rules always eliminate the smallest capacity sites regardless of their positioning. The resulting SITING solution may retain and fully use large sites far removed from most unit set GDPs while discarding small, well-positioned sites. Thus, the solution with zero redundant sites allowed will indeed use the fewest number of sites, but the solution may perform poorly relative to SITING Objective 1 because some allowed sites may be very badly positioned. The user can selectively eliminate specific sites by altering input to reset their storage capacity to zero.

CHAPTER 7

ILLUSTRATIVE TRADEOFF ANALYSIS

7-1. PURPOSE. This chapter uses assessment results from example cases with a common basic scenario to illustrate SITING tradeoffs. The example cases differ in user-set scenario options and project dispersion options. MOEs assessing how well each solution meets SITING objectives are compared over the example cases. Rationales for explaining these differences are also developed.

7-2. BASIC SCENARIO ATTRIBUTES. A basic example scenario was constructed with the following attributes:

a. Timeframe was 8 years.

b. About 500 unit sets of varying sizes were present

c. The unit sets were grouped into 14 projects.

d. There were 23 storage sites.

7-3. EXAMPLE CASES. SITING was executed for the cases shown in Table 7-1. The project dispersion option and the allocation option entries in the table refer to the choice of project dispersion option and allocation scenario option from the SITING menu-driven option input described in paragraph 4-2. All cases use exactly the same input data in the basic scenario with the cited attributes, except for the variations described in Table 7-1.

Table 7-1. Allocation Cases Analyzed by SITING

Case	Proj dispersion option	Allocation option
1	Ignore dispersion	Available storage
2	Ignore dispersion	Unconstrained storage
3	Reduce dispersion	Available storage
4	Reduce dispersion	Unconstrained storage

The interactive SITING menu inputs used for the cases shown in Table 4-3 are:

a. Allocation Scenario Option. Example cases 1 and 3 use the available storage case allocation scenario option, in which each storage site capacity is as generated by basic scenario input data. Cases 2 and 4 use the unconstrained storage case allocation scenario option, in which each storage site can hold an unlimited number of unit sets.

b. Project Dispersion Option. Cases 1 through 2 use the ignore dispersion option. Cases 3 and 8 use the reduce dispersion option.

c. Redundant Sites. All allocation cases allow all available sites to be used. Therefore, no redundant site is excluded.

7-4. COMPARATIVE CASE RESULTS. Tables 7-2 through 7-4 show assessment results for all cases of Table 7-1. Each table shows the value of an MOE which relates to one of the SITING objectives. The tabulated MOEs are averaged over all unit set solution storage allocations for each year of each case. The tabulated MOEs are described in Chapter 3 and Appendix E. In the context of the cases assessed, these are:

a. Average distance (km) of a stored unit set from its solution storage site to its unit set GDP location. This assesses the siting plan relative to SITING Objective 1. Results for this MOE are shown in Table 7-2.

b. Average number of sites used in storing an average project. This assesses the siting plan relative to SITING Objective 3. Results for this MOE are shown in Table 7-3.

c. The fraction of unit sets which had to be resited in each year, i.e., the fraction of unit sets in each year whose solution storage site is different from that unit's solution site in the previous year. This assesses the siting plan relative to SITING Objective 2. Results for this MOE are shown in Table 7-4.

Table 7-2. Assessments of Average Stored Unit Set Distance (km) to GDP

Year	Case			
	1	2	3	4
1	109	79	138	110
2	101	80	157	112
3	100	80	156	112
4	101	80	157	113
5	102	79	163	115
6	101	79	161	115
7	101	79	160	115
8	97	79	142	115

Table 7-3. Assessments of Average Number of Sites Needed to Store a Project

Year	Case			
	1	2	3	4
1	5.1	4.7	1.5	1.0
2	4.6	4.2	1.2	1.0
3	4.4	4.1	1.2	1.0
4	3.8	3.6	1.2	1.0
5	4.1	3.9	1.2	1.0
6	4.6	4.2	1.1	1.0
7	4.8	4.5	1.1	1.0
8	4.8	4.5	1.1	1.0

Table 7-4. Assessments of Fraction of Unit Sets Resited Each Year

Year	Case			
	1	2	3	4
1	.90	.89	.88	.92
2	.17	.00	.49	.00
3	.01	.00	.02	.00
4	.03	.00	.01	.00
5	.06	.00	.08	.00
6	.04	.00	.04	.00
7	.02	.00	.02	.00
8	.08	.00	.15	.00

7-5. TRADEOFF RATIONALE. The following rationale regarding the MOE tradeoffs with varying allocation scenario options applies:

(1) Available Storage versus Unconstrained Storage Tradeoff. The available storage option is the only one that yields a realistic siting solution. However, the siting plan produced by the unconstrained storage option shows how the storage sites should be "sized" in order to store each and every unit set at its optimally positioned (relative to SITING problem Objective 1) storage site. Under the unconstrained storage option, SITING will tend to store unit sets significantly closer to their unit set GDPs than under the available storage option.

(2) Ignore Dispersion versus Reduce Dispersion Tradeoff. With ignore dispersion option set, the resulting solution siting plan will emphasize SITING problem Objective 1 (positioning close to unit set GDP), but will not treat Objective 3 (reduction of project dispersion). That solution plan will likely have many projects stored over several sites. If the reduce dispersion option is used, the solution siting plan may greatly reduce the number of sites over which an average project is dispersed, but this solution will be worse (than the ignore dispersion solution) relative to storing unit sets close to their GDPs. Objective 1 is traded off against Objective 3. Only experimentation can determine the extent of this tradeoff.

7-6. TRADEOFF ANALYSIS. The comparative results shown in Tables 7-2 and 7-3 clearly support the stated rationale as follows:

a. Available Storage versus Unconstrained Storage. The available storage case with ignore dispersion (Case 1) has its unit sets stored an average of about 20 km further from their GDPs than is the case (Case 2) for unconstrained storage under ignore dispersion (97-109 km versus 79-80 km). The difference between the cases with available storage/reduce dispersion (Case 3) and unconstrained storage/reduce dispersion (Case 4) is even greater (138-163 km versus 110-115 km).

b. Ignore Dispersion versus Reduce Dispersion. Both ignore dispersion cases (1-2) store a project, on average, over four to five separate sites, while the reduce dispersion cases (3-4) usually average close to a single site.

c. Tradeoff between Objective 1 and Objective 3. Although use of the reduce dispersion option did more nearly achieve Objective 3, it did not achieve Objective 1 nearly as well as the ignore dispersion option. For the available storage case under reduced dispersion (Case 3), unit sets were stored an average of about 50 km further from their GDPs than the corresponding case (1) under ignore dispersion (138-163 km versus 96-109 km).

7-7. RESITING TURBULENCE ANALYSIS. Table 7-3 shows that, in the vast majority of cases, resiting turbulence (SITING Objective 2) was very small after year 1. After year 1, fewer than 10 percent of stored units usually need to be resited. The few instances of larger fractions resited after year 1 are due primarily to the closing down of several sites preferred for allocations in earlier years. A site closure forces relocation of unit sets stored there. The large year 1 turbulence is due to the considerable mismatch between the input in-place siting (which was not generated by SITING) and the SITING solution siting plan produced for year 1.

APPENDIX A

STUDY CONTRIBUTORS

1. STUDY TEAM

a. Study Director

Mr. Walter J. Bauman, Force Systems Directorate

b. Other Contributors

Mr. J. Ted Ahrens

Mr. Howard E. Whitehead

2. PRODUCT REVIEW BOARD

Mr. Ronald J. Iekel, Chairman

Mr. John W. Warren

Mr. Robert M. Hart

APPENDIX B

REFERENCES/BIBLIOGRAPHY

REFERENCES

1. POMCUS Siting Alternatives (POMCUSITE) Study, CAA-SR-91-8, US Army Concepts Analysis Agency, September 1991 (SECRET)

2. POMCUS Unit Siting Alternatives (POMCUSITE) User's Manual, CAA-D-91-4, US Army Concepts Analysis Agency, September 1991 (UNCLASSIFIED)

3. POMCUS Unit Siting Alternatives (POMCUSITE) System Documentation, CAA-D-91-5, US Army Concepts Analysis Agency, September 1991 (UNCLASSIFIED)

BIBLIOGRAPHY

Morton, Curtis R., A Handbook of Problems and Their Solutions for the Practicing Logistician, Air Command and Staff College, April 1978

The Operations Research Problem Solver, Research and Education Association, 1983

Woolsey, Robert E. D., Operations Research for Immediate Application, A Quick and Dirty Manual, Harper and Row, 1975

Wagner, Harvey M., Principles of Operations Research, Prentice-Hall Inc., 1969

APPENDIX C

NUMERIC EXAMPLES OF ALGORITHMIC PROCESSING

C-1. INTRODUCTION. This appendix provides five numeric examples of the operation of the SITING algorithm to assist in the understanding of the algorithm operation. The examples are intended as simplified illustrations of the basic modes of operation of the algorithm. Three examples of allocation are provided using the ignore dispersion mode and two examples are provided using the reduce dispersion mode of operation. The examples are necessarily limited in scope and should be used in conjunction with the analytic description of the algorithm (Chapter 3) to ensure a rounded understanding of the allocation processes. Reference to the algorithm source code may be needed in some instances to resolve questions on processing procedure under complex siting situations. The examples are identified by number, processing conditions, and the page number in the following table.

Example	Processing Conditions	Page
1	Ignore Dispersion Mode - Goal Year	C-2
2	Ignore Dispersion Mode - Year 1	C-6
3	Ignore Dispersion Mode - Year 2	C-7
4	Reduce Dispersion Mode - Goal Year	C-9
5	Reduce Dispersion Mode - Year 1	C-14

C-2. UNITS OF MEASURE. Although units of measure are not indicated in the tabulated example results, the current SITING Model requires that all distances be measured in kilometers. This requirement, however, is caused by hard wiring in the SITING program code rather than any general methodological restriction. The unit of measure for site capacity is arbitrary as long as it is expressed in the same units of measure as the unit set sizes in input. The unit of measure for weight of a unit set is also arbitrary.

C-3. EXAMPLE 1: IGNORE DISPERSION MODE - GOAL YEAR. Assume the unit set and project priority data are as shown in Table C-1 and that no designated project or unit set allocation input for projects or unit sets is present. For tractability, this example will treat only unit sets 4, 5, and 7 in the table. Also assume that the site capacities and the sizes and weights of unit sets are as portrayed in Tables C-2 and C-3. Table C-4 portrays the distances between each unit set GDP/site combination. In Table C-4, unit set refers to the location of that unit set's GDP, and site refers to the location of the indicated storage site. Table C-5 then shows a site preference table for each unit set. This table ranks the sites according to their closeness to the GDP of the unit set. Table C-5 is constructed as a series of such site preference lists, one list for each unit set. For example, the site preference list for unit set 5 consists of site 2 and site 1, in that order. Table C-6 portrays the results of each allocation step in processing the indicated unit sets in the goal year. Note, in Table C-6, that the unit sets 4, 5, and 7 are processed in the order 5, 4, 7, which is in increasing order of their combined priority (Table C-1). The algorithm then processes these unit sets as follows:

1. From the site preference list in Table C-5, the preferred (closest to GDP) storage site for unit set 5 is site 2. Since site 2 capacity of 300 is at least as large as the size (storage requirement) of unit set 5 (size = 100), SITING allocates unit set 5 to site 2. The first line of Table C-6 also shows the space occupied/space unoccupied status of each site after this first allocation.

2. From the site preference list in Table C-5, the preferred (closest to GDP) storage site for unit set 4 is site 2. Since site 2 space unoccupied at this time (after allocation of unit set 5) is 200, which is less than the size (storage requirement) of unit set 4 (= 800), unit 4 can not fit at its preferred site. SITING therefore tries to allocate it to the next most preferred site in its site preference list. This is site 1. The space now unoccupied at site 1 is 1500, which exceeds the unit set 4 size of 800, so SITING allocates unit set 5 to site 1. The second line of Table C-6 also shows the space occupied/space unoccupied status of each site after this allocation.

3. From the site preference list in Table C-5, the preferred (closest to GDP) storage site for unit set 7 is site 2. Since current site 2 unoccupied space of 200 is at least as large as the size (storage requirement) of unit set 7 (= 200), SITING allocates unit set 7 to site 2. The last line of Table C-6 also shows the space occupied/space unoccupied status of each site after this allocation.

Note that unit set weight is not used to determine allocations in the above example. In this case, it is used only in assessment, e.g., a useful MOE for a SITING solution is the ton-km movement requirement of an allocated unit set from its allocation site to its GDP. This MOE is the product of the unit set weight and the distance from allocation site to GDP.

The allocations computed for the goal year then serve as allocation goal sites over the entire timeframe. That is, when processing every other year in the timeframe, SITING will attempt to allocate each unit set to its goal site, thus pursuing Objective 2 (turbulence), as well as Objective 1 (unit

set siting), of the SITING problem. Note that the above example algorithm in the goal year treated only SITING Objective 1. Since the goal year allocations are the fulcrum for all resiting, Objective 2 is not treated that year. Since the example was for a scenario which ignored project dispersion, SITING Objective 3 (project dispersion) is not treated here.

Table C-1. Unit Set/Project Priorities

Unit set	Project	Unit set priority	Project priority	Combined priority
1	1	25	3	30025
2	1	156	3	30156
3	1	2	3	30002
4	2	2,990	1	12990
5	2	10	1	10010
6	3	10	2	20010
7	3	2	2	20002

NOTE: The combined priority is calculated as follows. The largest unit set priority is 2990. The smallest power of 10 exceeding this is 10000. The combined priority is then computed as:

$$10000 \times [\text{Project priority}] + [\text{Unit set priority}]$$

For unit set 1 in the table, the computed combined priority is then 3 x 10000 + 25 = 30025. The other values are similarly computed. Observe that the numeric increasing order of the unit sets according to the combined priorities has the unit sets ordered first according to their project's priority and secondly according to their unit set priority.

Table C-2. Site Capacities in Goal Year

Site	Capacity
1	1500
2	300

Table C-3. Unit Sizes and Weights

Unit set	Size	Weight
5	100	50
4	800	200
7	200	90

Table C-4. Distance from Unit Set GDP to Each Site

Unit Set	Site	Distance
5	1	999
5	2	111
4	1	999
4	2	111
7	1	999
7	2	111

Table C-5. Site Preference Rankings for Each Unit Set

Unit Set	Site	Distance
5	2	111
5	1	999
4	2	111
4	1	999
7	2	111
7	1	999

Table C-6. Allocation in Goal Year - Dispersion Ignored

Unit set	Allocation site	Postallocation space occupied		Postallocation space unoccupied	
		Site 1	Site 2	Site 1	Site 2
5	2	0	100	1500	200
4	1	800	100	700	200
7	2	800	300	700	0

C-4. EXAMPLE 2: IGNORE DISPERSION MODE - YEAR 1. Assume that input data shown in Tables C-1, C-3, C-4, and C-5 of Example 1 apply, but that the site capacities for year 1 are changed to that shown in Table C-7. Again, assume that no designated project or unit set allocation input is present. Also assume that the goal year allocations of Table C-6 apply. Table C-8 portrays the sequenced results of the allocation process for these unit sets in year 1. The algorithm is:

1. Unit sets are processed in order of combined priority, i.e., unit sets 5, 4, and 7.

2. Unit set 5 (size = 100) is allocated to its goal site, site 2, since there is sufficient unoccupied space (=150) there.

3. Unit set 4 (size = 800) is allocated to its goal site, site 1 since there is sufficient unoccupied space (=1200) there.

4. Unit set 7 will now not fit at its goal site (site 2) because the unoccupied space at site 2 (= 50) is less than the unit set 7 size (= 200).

Therefore, unit set 7 is allocated to site 1 (available space = 800), which is the next most preferred site for it in its site preference table (in Table C-5).

Table C-7. Site Capacities in Year 1

Site	Capacity
1	1200
2	150

Table C-8. Allocation in Year 1 - Dispersion Ignored

Unit set	Allocation site	Postallocation space occupied		Postallocation space unoccupied	
		Site 1	Site 2	Site 1	Site 2
5	2	0	100	1200	50
4	1	800	100	400	50
7	1	1000	100	200	50

C-5. EXAMPLE 3: IGNORE DISPERSION MODE - YEAR 2

Again, assume that Tables C-1, C-3, C-4, and C-5 of Example 1 apply, but that the site capacities for year 2 are as shown in Table C-9. Again, assume that no designated project or unit set allocation input is present. Also assume that the goal year allocations of Table C-6 continue to apply, as do the year 1 allocations of Table C-8. Table C-10 shows in-place sites and goal sites. The allocation sites of Table C-8 are the in-place sites of Table C-10. The allocation sites in Table C-6 are the goal sites in Table C-10. Table C-11 portrays the sequenced results of the allocation process for these unit sets in year 2. The unit set combined priority sequence is unit set 5, 4, and 7 in that order. The algorithm is:

1. Table C-10 shows that only unit sets 5 and 4 are in place at their goal sites. These are processed first in combined priority order as follows:

   a. Unit set 5 is allocated to its goal site, site 2, since there is sufficient unoccupied space (=300) there.

   b. Unit set 4 is allocated to its goal site, site 1, since there is sufficient unoccupied space (=1300) there at that time.

   c. All other unit sets (of lower combined priority) which are in place at their goal sites would now be processed ahead of those that are not in place at their goal sites, even if any of the latter has a higher combined priority than the former. In this stylized example, assume no other unit sets.

2. Having finished allocating the unit sets in place at goal sites, SITING now processes, in order of combined priority, all other unit sets, i.e., those not allocated in (1). In this example, this set consists only of unit set 7. Unit set 7 is allocated to its goal site, site 2, since there is room (= 200) there at that time.

Table C-9. Site Capacities in Year 2

Site	Capacity
1	1300
2	300

Table C-10. Siting Status at Start of Year 2

Unit set	In-place site	Goal site
5	2	2
4	1	1
7	1	2

Table C-11. Allocation in Year 2 - Dispersion Ignored

Unit set	Allocation site	Postallocation space occupied		Postallocation space unoccupied	
		Site 1	Site 2	Site 1	Site 2
5	2	0	100	1300	200
4	1	800	100	500	200
7	2	800	300	500	0

C-6. EXAMPLE 4: REDUCE DISPERSION MODE - GOAL YEAR. This is an entirely different example from those in Examples 1 through 3. Assume that no designated project or unit set allocation input is present. Let Table C-12 define unit set and project priorities. Let Table C-13 define site capacities. Let Table C-14 define unit set sizes and weights. Let Table C-15 show distances between each unit set GDP and each site. Then Table C-16 is the site preference table for unit sets. In Table C-16, unit set 1, 2, 3 means that the line applies to each of unit sets 1, 2, and 3.

1. Table C-17 is the result of constructing the site preference lists for the projects. It consists of a series of site preference lists, one for each project, P, with sites ranked according to increasing closeness to project P GDP measures for the site. Thus, for example, the order of site preference for project 2 is: site 2, site 3, and site 1. For project 2, consisting of unit sets (US) 4 and 5, using Tables C-14 and C-15, the algorithm calculation of closeness to GDP for site 3 (paired with project 2) in the site preference list for project 2 is [(weight of US 4) x (distance of US 4 GDP to site 3) + (weight of US 5) x (distance of US 5 GDP to site 3)]/[combined weight of US 4 and US 5] = (10 x 300 + 100 x 300)/110 = 300. This is the entry on the fifth line of Table C-17.

2. The order of increasing project priority is 2, 3, and 1. This is also the order of processing. In the project allocation phase SITING tries to allocate these intact to their most preferred sites in Table C-17. The processing is at each stage is shown in Table C-18:

a. Project 2 can fit at site 2, its most preferred site (see Table C-17), so it is allocated there.

b. Project 3 also has site 2 as its most preferred site, but cannot fit there because its size (= 200) exceeds the space now available (= 60) at site 2. The next most preferred site for project 3 is site 3 (unoccupied space = 260). Project 3 can fit there and is allocated there.

c. Project 1 cannot fit intact at any site. It is too large (size = 400). It is not allocated in this phase.

3. In the unit set allocation phase, the remaining unallocated projects are processed in project priority order. Only project 1 is to be allocated here. Each unit set of the project is processed in order of unit priority (Table C-12). Thus the order in the example is unit sets 2, 3, and 1. The processing sequence of these unit sets, summarized in Table C-19, is:

a. SITING constructs the shortlist of sites for allocation of project 1. From Table C-17, the site preference order for project 1 is site 1, site 2, and site 3. SITING accumulates the current space unoccupied at these sites (see the last line of Table C-18) and determines that sites 1 and 2 (accumulated unoccupied space = 360 + 60 = 420) probably can contain the remaining unallocated unit sets of project 1. These two sites, in the order listed, comprise the shortlist for allocating project 1.

b. The site on the project 1 shortlist which is closest to the unit set 2 GDP is site 1. Site 1 has sufficient unoccupied space (= 360) for unit set 2 (size = 50), so it is allocated there.

c. Unit set 3 (size = 300) can fit at the site on the shortlist which is closest to its GDP (site 1 with space available = 60), so it is allocated there.

d. Unit set 1 (size = 50) cannot fit at the site on the project 1 shortlist (site 1 with unoccupied space = 10) that is closest to its GDP. It does fit at its next most preferred site (site 2) on the shortlist and is allocated there.

Table C-12. Unit Set/Project Priorities

Unit set	Project	Unit set priority	Project priority	Combined priority
1	1	36	3	3036
2	1	12	3	3012
3	1	24	3	3024
4	2	118	1	1118
5	2	54	1	1054
6	3	24	2	2024
7	3	612	2	2612

Table C-13. Site Capacities in Goal Year

Site	Capacity
1	360
2	260
3	260

Table C-14. Unit Set Sizes and Weights

Unit set	Size	Weight
1	50	10
2	50	10
3	300	50
4	100	10
5	100	100
6	100	10
7	100	100

Table C-15. Distance From Unit Set GDP to Each Site

Unit set	Site	Distance
1,2,3	1	200
1,2,3	2	300
1,2,3	3	700
4,5,6,7	1	800
4,5,6,7	2	200
4,5,6,7	3	300

Table C-16. Site Preference Rankings for Each Unit Set

Unit set	Site	Distance
1,2,3	1	200
1,2,3	2	300
1,2,3	3	700
4,5,6,7	2	200
4,5,6,7	3	300
4,5,6,7	1	800

Table C-17. Site Preference Rankings for Each Project

Project	Site	Closeness to GDP
1	1	200
1	2	300
1	3	700
2,3	2	200
2,3	3	300
2,3	1	800

Table C-18. Project Allocation in Goal Year - Reduced Dispersion

Project	Allocation site	Postallocation space occupied			Postallocation space unoccupied		
		Site 1	Site 2	Site 3	Site 1	Site 2	Site 3
2	2	0	200	0	360	60	260
3	3	0	200	200	360	60	60

Table C-19. Unit Set Allocation in Goal Year - Reduced Dispersion

Unit set	Allocation site	Postallocation space occupied			Postallocation space unoccupied		
		Site 1	Site 2	Site 3	Site 1	Site 2	Site 3
2	1	50	200	200	310	60	60
3	1	350	200	200	10	60	60
1	2	350	250	200	10	10	60

C-7. EXAMPLE 5: REDUCE DISPERSION MODE - YEAR 1. We will walk through year 1 allocations of following on the allocations of Example 4. Assume that no designated allocation input is present. Assume that the goal year Tables C-12, C-14, C-15, C-16, and C-17 apply also to year 1. However, the site capacities for year 1 are changed (from the goal year) and are shown in Table C-20. Table C-22 portrays the sequenced results of the allocation process for these unit sets in year 1. The algorithm for processing Example 5 in year 1 is:

1. The project allocations of Table C-18 define the project goal sites. These are sites 2 and 3 for projects 2 and 3, respectively. The unit set allocations of Table C-19 define the unit set goal sites. These are sites 1, 1, and 2 for unit sets 2, 3, and 1, respectively.

2. Table C-21 shows the sequence and results of processing in the project allocation phase. Project priority order (and the order of processing) is project 2, 3, and 1.

a. Since there is insufficient space for all of project 2 (size = 200) to fit at its goal site (site 2), and since year 1 has no in-place site assignments, SITING allocates it to the most preferred available site on the site preference list for project 2 (Table C-17). The most preferred site (site 2) has insufficient space, but the next most preferred site (site 3) can fit all of project 2, so it is allocated there.

b. Since project 3 can fit intact at its goal site (site 3), it is allocated there.

c. Project 1 is too large to fit at any site now, so it is not allocated in this phase, but will be treated in the unit set allocation phase.

3. Table C-22 shows the sequence and results of processing in the unit set allocation phase. Only project 1 is being processed in this phase. The unit priority order of unit set processing in project 1 is unit set 2, unit set 3, and unit set 1. The algorithm is:

a. The ordered site shortlist for all of project 1 clearly consists of site 1 and site 2 (in that order) since their total unoccupied space at this time is 240 + 170 = 410, which exceeds the project 3 size of 400.

b. Unit sets in place at their goal sites are processed first, but, since in-place positions are not defined for year 1, nothing is done here in this case.

c. Remaining unallocated unit set processing is as follows:

1. Unit set 2 can fit at its goal site (site 1), so it is allocated there.

2. Unit set 3 (size = 300) cannot fit at its goal site (site 1). There is no in-place status yet, so SITING tries to allocate it to the most preferred site on the project 1 shortlist. It will not fit at any site on the shortlist. Therefore, SITING tries to allocate it to the most preferred available site on its site preference list (Table C-16). Neither site 1 nor site 2 has room, but the next site on the list, site 3, has sufficient unoccupied space, so unit set 3 is allocated there.

3. Unit set 1 can fit at its goal site (site 2), so it is allocated there.

Table C-20. Site Capacities in Year 1

Site	Capacity
1	240
2	170
3	725

Table C-21. Project Allocation in Year 1 - Reduced Dispersion

Project	Allocation site	Postallocation space occupied			Postallocation space unoccupied		
		Site 1	Site 2	Site 3	Site 1	Site 2	Site 3
2	3	0	0	200	240	170	525
3	3	0	0	400	240	170	325

Table C-22. Unit Set Allocation in Year 1 - Reduced Dispersion

Unit set	Allocation site	Postallocation space occupied			Postallocation space unoccupied		
		Site 1	Site 2	Site 3	Site 1	Site 2	Site 3
2	1	50	0	400	190	170	325
3	3	50	0	700	190	170	25
1	2	50	50	700	190	120	25

APPENDIX D

ERROR MESSAGES GENERATED DURING SITING EXECUTION

D-1. ERROR MESSAGES GENERATED. During SITING execution, messages indicating any errors found are written to the screen. These errors may be fatal or not fatal. A fatal error will cause processing to terminate and will invalidate the execution results. The following is an explanation of error messages and warnings that SITING is programmed to find and display. In the message examples below, N1, N2, N3, and N4 represent numbers in the error messages. These numbers would be generated and determined by the program.

D-2. INPUT DATA ERROR MESSAGES. The following messages indicate peculiarities, discrepancies, and inconsistencies in the SITING input data.

(1) ** WARNING: FILE9 - YEAR N1 IS NOT = N2 FROM FILE8

Meaning: This means that the year data blocks of the site definition input data File (File9.txt) are not in sync with the year data blocks of the unit definition input data file (File8.txt). The values N1 and N2 are numeric year IDs of year data blocks in the same position in the two files. The n-th year data block of each file should have the same numeric year ID. This is not a fatal error and will not in itself abort the run.

(2) FATAL ERROR: NR SITES = N1 FROM FILE9 FOR YEAR N2 EXCEEDS 30.

Meaning: The number of distinct storage sites over all years, as defined in the site definition input data file (File9.txt), plus the two fictitious overflow sites (sites -01 and -99) exceeds 30. No more than 30 storage sites are allowed over all years. This is a fatal error and aborts the run. (The value 30 is the setting of a program parameter called MXSITE.)

(3) ** FATAL ERROR: UNIT N1 IN YEAR N2 HAS UNIT SET ID EXCEEDING 2000.

Meaning: A unit set defined in the year N2 data block of the unit definition file (File8.txt) has the numeric unit set ID N1, which exceeds 2000. All numeric unit set IDs must be no more than 2000. (The value 2000 is the setting of a program parameter called MXHOLD.)

(4) UNIT SET N1 IN YR N2 IGNORED: SIZE = WT = 0

Meaning: This is a nonfatal warning that the unit set defined in the unit definition input data file (File8.txt) with numeric ID N1 in the year N2 data block of that file has a defined storage requirement of zero and a defined total weight of zero. The unit set will be ignored during processing because it is empty.

(5) UNIT SET N1 IN YR N2 IGNORED - NO LAT, LONG LOCATION

Meaning: This is a nonfatal warning that the unit set defined in the unit definition input data file (File8.txt) with numeric ID N1 in the year N2 data block of that file has a defined location at 0 latitude. The program assumes that the location is in error and was probably omitted. The unit set is ignored during processing.

(6) ** FATAL ERROR: N1 = NR OF UNITS IN YR N2 EXCEEDS 1000

Meaning: This is a fatal error. More than 1000 different unit sets were in year block N1 in the unit definition input file (File8.txt). At most, 1000 unit sets are allowed in any one year and over all years. With this error, the run aborts. (The value 1000 is the setting of a program parameter MXUNIT.)

(7) ** WARNING: FILE 10- YEAR N1 IS NOT = N2 FROM FILE 8

Meaning: This means that the year data blocks of the project definition input data file (File10.txt) are not in sync with the year data blocks of the unit definition input data file (File8.txt). The values N1 and N2 are numeric year IDs of year data blocks in the same position in the two files. The n-th year data block of each file should have the same numeric year ID. This is not a fatal error and will not in itself abort the run.

(8) FATAL ERROR: PROJECT N1 IN FILE 10 FOR YEAR N2 HAS ID EXCEEDING 25

Meaning: A project set defined in the year N2 data block of the project definition file (File10.txt) has the numeric project ID N1, which exceeds 25. All numeric project IDs must be no more than 25. (The value 25 is the setting of a program parameter called MXPKG.) This is a fatal error and will cause a run abort and invalidation of any output.

(9) FATAL ERROR: PROJECT N1 IN YR N2 HAS MORE THAN 200 UNITS IN IT

Meaning: A project set defined in the year N2 data block of the project definition file (File10.txt) has been defined to have in excess of 200 unit sets comprising it. No one project is allowed to have more than 200 unit sets. (The value 200 is the setting of a program parameter called MXLIST.) This is a fatal error and will cause a run abort and invalidation of any output.

(10) ERROR: UNIT N1 IN PROJECT N2 FOR YEAR N3 IS NOT DEFINED IN FILE8

Meaning: The unit set with ID N1 defined in the data sub-block for project N2 in the year N3 data block in the project definition input data file (File10.txt) was never defined in the unit definition input data file (File8.txt) i.e., a unit set with ID N1 in year N3 was never defined in

File8.txt. This unit will be ignored during processing and allocations will proceed, if possible. Therefore, this error is not fatal by itself.

(11) FATAL ERROR: UNIT N1 IN YR N2 IS NOT ASSIGNED A PROJECT ID IN FILE10

Meaning: The unit set defined with numeric ID N1 in year N2 in the unit definition input data file (File8.txt) was never defined as belonging to any project definition input data file (File10.txt). Each unit set must belong to exactly one project. This is a fatal error and will cause a run abort and invalidation of any output.

(12) FATAL ERROR: IN-PLACE SITE N1 IS NOT DEFINED IN FILE9

Meaning: The numeric ID, N1, of a site in a site block of the in-place definition input data file (File11.txt) was never defined in the site definition file (File9.txt). This is a fatal error and will cause a run abort and invalidation of any output.

(13) FATAL ERROR: SITE N1 HAS MORE THAN 200 IN-PLACE UNITS

Meaning: The site with numeric ID N1 in the in-place definition input data file (File11.txt) has more than 200 numeric unit set IDs in its associated site data block. During SITING processing, the list of in-place unit sets at each site may be slightly augmented. At most, 200 in-place unit sets are allowed at any one site in the augmented list. (The value 200 is the setting of a program parameter, MXLIST.) This is a fatal error and will cause a run abort and invalidation of any output.

(14) ERROR: UNIT N1 IN-PLACE AT SITE N2 IS NOT DEFINED IN FILE 8

Meaning: The numeric ID N1 of a unit set in the site data block for site N2 in the in-place definition input data file (File11.txt) was not defined in the unit definition file (File8.txt). This error is nonfatal. SITING will continue normally, but will ignore unit set N1 in all processing.

(15) FATAL ERROR: N1 = NR OF UNITS THRU YR N2 EXCEEDS 1000

Meaning: This is a fatal error. More than 1000 different unit sets were defined over all 8 years in the unit definition input file (File8.txt). At most 1000 unit sets are allowed in any one year and over all years. (The value 1000 is the setting of a program parameter MXUNIT.) This is a fatal error and will cause a run abort and invalidation of any output.

(16) FATAL ERROR: ALLOC SITE N1 IN YEAR N2 DOES NOT EXIST NEXT YEAR

Meaning: The site defined in the year N2 data block of the site definition input data file (File9.txt) is not also defined in the year (N2+1) data block

of that file. A site present in any year must be defined in each year in the site definition file. If the site is not being used in a year, it must be assigned a zero storage capacity in that year block in the site definition file. This is a fatal error and will cause a run abort and invalidation of any output.

(17) ** WARNING - UNIT N2 IN YR N2 IS NOT PRESENT IN THE FINAL YEAR

Meaning: The unit set defined with numeric unit set ID N1 in the year N2 data block of the unit definition input data file (File8.txt) is not also defined in the year 8 data block of the unit definition input file. Since year 8 is the goal year which is intended to be a basis for setting siting goals for unit sets in other years, the unit identified in this message cannot be assigned such a goal. For such a unit, SITING assigns, as a goal site, the closest available site to the unit set's GDP in the first year it appears. Since such cases should be rare, the user is informed.

(18) FATAL ERROR: SITE N1 HAS N2 UNITS IN-PLACE - EXCEEDING 200

Meaning: SITING does slight augmentation of the list of in-place units read in from the in-place definition input data file. This error occurs if, in the augmented list, the site with numeric ID N1 has more than 200 in-place units assigned to it. If message (13) above did not occur, then the augmentation induced the error. In such a case, the user should gradually reduce the number of in-place units assigned to site N1 in the in-place definition input file (File11.txt) until the error disappears. No more than 200 in-place unit sets are allowed at any one site in the augmented in-place list. (The value 200 is the setting of a program parameter, MXLIST.) This is a fatal error and will cause a run abort and invalidation of any output.

D-3. INFEASIBILITY AND OVERFLOW ERRORS. If the following error message occurs, the user has defined a problem that has no solution because the total storage available, as defined in the site definition input data file (File9.txt), is insufficient to store all the unit sets that must be allocated in some year.

ERROR: INSUFFICIENT TOTAL SITE CAPACITY IN YR N1
AT LEAST N2 MORE STORAGE IS NEEDED.

Meaning: SITING summed all of the defined (in the site definition file (File9.txt) site capacities for year N1. SITING also summed all of the defined (in the unit definition file, File8.txt) unit set sizes (storage requirements) in year N2. The summed site capacity was N2 less than the summed total storage requirements of the unit sets. Therefore, the problem is infeasible unless at the summed site capacity is increased to a level in excess of the summed unit set sizes. Since unit sets are not "broken" by SITING allocations, i.e., they are allocated intact, slightly more than N2 may be needed to ensure feasibility. SITING deals with this problem by using two fictitious overflow storage sites, denoted as site -99 and site -01. The

user must check if any allocations were made to overflow sites -99 or -01. This check is the test of infeasibility due to insufficient storage.

D-4. OPERATIONAL ERROR MESSAGES. SITING had to be constrained in certain operating parameters for the model to fit on a PC. If, during the course of a solution, the status of certain algorithm activities "goes out of bounds," the user is informed via the following messages.

(1) ERROR - YR N1 NR ALLOC TO SITE N2 EXCEEDS 200 - WITH UNIT N3

Meaning: The above message says that the total number of unit sets allocated by SITING to site with numeric ID N2 (as defined in input File9.txt) in year N1 exceeds 200. However, at most 200 unit sets can be allocated to a site. Otherwise, the algorithm has a high likelihood of not functioning properly because internal memory constraints are broken. The value 200 is the setting of a program parameter MXLIST. The resulting solution with this error message should be treated as invalid even if the run does not abort. The associated siting problem is therefore unsolvable with the current configuration of SITING.

(2) WARNING: YEAR N1 ALLOCATIONS ARE INVALID BECAUSE NUMBER OF UNITS ALLOCATED TO A SITE EXCEED 200

Meaning: The meaning is essentially the same as for message (1).

(3) WARNING: YR N1 SITE N2 AT LIMIT OF 200 UNITS ALLOCATED - SITE NOW CLOSED

Meaning: The above message says that the total number of unit sets allocated by SITING to site with numeric ID N2 (as defined in input File9.txt) in year N1 exceeds 200. The value 200 is the setting of a program parameter MXLIST and is restricted because of PC memory constraints. The message also says that SITING tried to stop all further unit set allocations to site N2 as soon as this was detected (i.e., SITING "closed" the site). In most cases, this will prevent in excess of 200 unit sets being allocated to site N2. If so, messages (1) and (2) above will not appear and the solution is valid. However, the algorithm used with the reduce dispersion option may not succeed in closing site N2 soon enough to prevent more than 200 unit sets being allocated there.

D-5. ERRORS IN DESIGNATED ALLOCATIONS. If the user constructed and used the optional designated allocations input file (File7.txt), the following messages indicate status, inconsistencies, and/or errors in operation with that file.

(1) ** FILE7 IS INPUT IN THIS RUN **

Meaning: This informs the user that an input designated allocations File7.txt is being read. This is useful in case the user had forgotten to erase an old File7.txt and really did not want it in the current run.

(2) DESIGNATED ALLOCATION LIST TRUNCATED - EXCEEDS 50 ITEMS

Meaning: The user has tried to input more than 50 records in file File7.txt. SITING reads the first 50 records, ignores all others, and writes the above message. At most, 50 records (program parameter MXFL) are allowed in this file.

(3) UNABLE TO DESIGNATE PROJECT N1 IN YEAR N2 TO SITE N3

Meaning: SITING cannot preallocate the designated project N1 intact to storage site N3 in year N2 even though File7.txt specified this. There can be several reasons for this. The specified project ID or site ID may not exist. It is also possible that there is insufficient available space at site N3 at allocation time to hold project N1. Since the designated allocations are done in order, the last projects processed for preallocation may be crowded out by earlier preallocations to the designated site.

(4) DESIGNATION OF PROJECT N1 IN YR N2 TO SITE N3

Meaning: This confirms that the specified project N1 was preallocated intact to site N3 in year N2.

(5) UNABLE TO DESIGNATE UNIT N1 IN YEAR N2 TO SITE N3

Meaning: SITING cannot preallocate the designated unit set with numeric ID N1 to storage site N3 in year N2 even though File7.txt specified this. There can be several reasons for this. The specified unit set ID or site ID may not exist. It is also possible that there is insufficient available space at site N3 at allocation time to hold unit set N1. Since the designated allocations are done in order, the last unit sets processed for preallocation may be crowded out by earlier preallocations to the designated site.

(6) DESIGNATION OF UNIT N1 IN YR N2 TO SITE N3

Meaning: This confirms that the specified unit set N1 was preallocated intact to site N3 in year N2.

APPENDIX E

OPTIONAL SITING FEATURES, INPUTS AND OUTPUTS

E-1. PURPOSE. This appendix explains how the user may extend SITING solution capability by using certain optional input files as specified below. These extensions are summarized in Table E-1. For each optional feature in Table E-1, the input file column names the optional input file which controls that feature. The features are described in more detail in the paragraphs describing the associated input file structure.

E-2. CAVEAT. All unit sets, storage sites, and projects in SITING must be labeled as integer numbers. All SITING output files are generated with unit sets, sites, and projects identified only in terms of the numeric identification (ID) labels used in SITING input. It is up to the user or his agent to develop any preprocessors needed to translate real-world names into SITING numeric labels and vice versa.

Table E-1. Optional SITING Features and Associated Input Files

Optional feature	Input file
Assess allocations reflected in in-place input File11	Onscreen menu
Change the number of years in the SITING problem timeframe from the default of 8 years	File15.txt
Generate MOE information on the SITING solution in outputs File14, File17, File20	File15.txt
Write descriptive column headings on output Files	File15.txt
Write project site preference list on output File18	File15.txt
Write error messages on File19 as well as on screen	File15.txt
Write detailed debug information on output File19	File15.txt
Change the frequency of writing outputs File16 and File21	File15.txt
Suppress writing of File12	File15.txt
Preallocate specified unit sets to specified sites before SITING algorithm operates	File7.txt
Preallocate specified unit sets to specified sites before SITING algorithm operates	File7.txt

E-3. OPTIONAL MENU-DRIVEN SCENARIO OPTION INPUT. In the first set of scenario option choices presented to the user, the on-screen choices presented to the user are :

```
CHOOSE ONE OPTION

1. AVAILABLE STORAGE CASE (ENTRY = 1)

2. UNCONSTRAINED STORAGE CASE (ENTRY = 2)

3. BASELINE CASE (ENTRY = 3)
```

When the baseline case option is selected, no siting plan allocation solution is generated by SITING. Instead, the "goodness" of the input in-place siting (as represented by input (File11.txt), using unit set characteristics for year 1, is assessed in light of how closely SITING objectives 1, 2, and 3 are met. Only one year (the first) is processed, and the only meaningful output files generated for this case are these which show assessment MOEs, namely File14.txt, File17.txt, and File20.txt.

E-4. OPTIONAL SITING INPUT FILES. The following input files, if constructed offline, can be used to extend SITING capabilities. They must be defined with the names File15.txt and File7.txt as given below and must be in ASCII format. If any of these optional files are not present on the input data disc, SITING will still operate properly under the default conditions described in the POMCUSITE manual (CAA-D-91-4). If one of these files is present in the directory containing the input data, it will automatically be read and processed by SITING.

a. Optional Input File15.txt. File15.txt is the scenario modifier file. It modifies defaults which are implicitly set if File15.txt is omitted. File15.txt consists of a single record with 10 parameter fields. Each parameter field overrides a default setting of the associated parameter. If File15.txt is used for input, all 10 parameters must be defined on it, even if some of them are the same as the default values. The structure of File15.txt is as follows (all data must be right justified in its columnar field):

Columns 3-5: the integer number of consecutive years to be processed in the timeframe. This must be between 1 and 8. The default is an 8-year timeframe. The only way to process a different timeframe is to set this input entry in File15.txt.

Columns 9-10: an integer indicator of whether to generate optional output File12.txt, the siting allocations list. If this entry is equal to zero, File12.txt will be omitted, otherwise it is generated. The SITING default is to generate File12.txt (default value = 1).

Columns 14-15: an integer indicator of whether to generate the standard output File16.txt, the site-site distance list, or a modified File16.txt. If this entry is equal to zero, only the standard File16.txt will be generated. If this entry is nonzero, only a modified File16.txt which includes text column headings head is generated. The SITING default is to generate only a standard File16.txt (default value = 1).

Columns 19-20: an integer indicator of whether to generate optional output File17.txt, the MOE summary and output File14.txt, the site fill report. If the user wishes to have SITING assess the generated solution siting plan, this entry must be set to a nonzero value. If this entry is equal to zero, neither File17.txt or File14.txt will be generated; otherwise, both of these output files are generated. This is equivalent to omitting the assessment phase of SITING. The SITING default is to omit File17.txt and File14.txt (default value = 0).

Columns 24-25: an integer indicator of whether to generate optional output File18.txt, the project site preference lists. If this entry is equal to zero, File18.txt will be not generated, otherwise it is generated. The SITING default is to omit File18.txt (default value = 0).

Columns 29-30: an integer indicator of whether to write all SITING-generated on-screen error messages on File19.txt as well as on the screen. If this entry is zero, the error messages are not written into File19.txt, otherwise they are. The SITING default is not to copy the onscreen error messages onto File19.txt (default value = 0).

Columns 34-35: an integer indicator of whether to generate optional output File20.txt, the detailed siting characteristics list. If this entry is equal to zero, File20.txt will be omitted; otherwise, it is generated. The SITING default is to omit File20.txt (default value = 0).

Columns 39-40: an integer indicating the year for which the unit set site preference list output File (File21.txt) is to be generated. In order to generate anything, this value must be between 1 and 9 or equal to 99. If the value, N, is between 1 and 8, File21.txt is written only for year N. Either a value of 9 or a value equal to the goal year will write it only for the goal year. A value of 99 writes the file for all years in the timeframe. All other values cause no output File21.txt to be generated. The SITING default setting is 9 (generate File21.txt only in the goal year).

Columns 44-45: an integer indicator of whether to add column headings to outputs File14.txt, File17.txt, File18.txt, File20.txt, and File21.txt. If this entry is equal to zero, no column headings are included in any of the cited output files. If this entry is nonzero, brief descriptive column headings are added to the cited output files. This entry has no effect on the content or format of the data contained in the cited output files. Omission of headings facilitates analysis of the output with a data base processor. The SITING default is to omit headings (default value = 0).

Columns 49-50: an integer indicator of whether detailed debug information is to be written to File19.txt. If this value is set to N1, where N1 is a year in the timeframe (between 1 and 8), then debug information is

written only for that year's processing. If this value is set to 99, debug information is written for processing in all years. Any other value suppresses writing of this debug information. This information can be extremely massive in its size (about 1 megabyte per year debugged), so the user is cautioned to avoid requesting it unless he knows how much he will get and how it can be used. Only examination and firm understanding of the SITING source program code can justify the use of this feature since it is not in friendly English. The SITING default is to not write this debug information (default entry equal to 0).

b. Optional User-designated Allocations. This feature is controlled by optional input File7.txt. It enables a user to specify preallocation of selected projects and/or unit sets to specific sites before the normal SITING allocations are begun. With this feature, a user may specify the following allocation overrides for selected project and/or unit sets:

(1) Designated Project Allocations. If this option is specified, the user wishes to preallocate specific projects intact at specific sites so that the algorithm will allocate "around" them. The user identifies, for each such project, the year(s) in which the designated allocation should occur and the site where the entire project should be allocated. The specified projects are preallocated in order of project priority if they can fit. If a project does not fit at its designated site, the algorithm allocates it normally.

(2) Designated Unit Set Allocations. If this option is used, the user wishes to preallocate specific unit sets at specific sites so that the algorithm will allocate around them. The user identifies, for each such unit set, the year(s) in which the designated allocation should occur and the site where the designated unit set should be allocated. The specified unit sets are preallocated in order (designation) input if they can fit. If a unit set does not fit at its designated site, the algorithm allocates it normally.

Optional input File7.txt consists of a series of records. Each record specifies exactly one designated project allocation and/or one designated unit set allocation. The designated project allocations are processed in order of project priority. The designated unit set allocations are processed in order of their input in this file. If a designated allocation cannot be done, it will be ignored, and the SITING algorithm will allocate the named unit set/project as if the user had never specified the designated placement. The user must have no more than 50 records in this file. (The value 50 is the setting of a program parameter MXFL.) The structure of each record is as follows. All data must be right justified in the columned fields.

Columns 1-5: the integer numeric project ID (from input File10.txt) of the project to be allocated by designation.

Columns 6-10: an integer indicating the year the allocation of the indicated project is to take effect. This value must be between 1 and 8 or must be equal to 99. If the value, n, is between 1 and 8, the designated allocation will take effect, if possible, in year n. If the value is 99, the designated allocation will take effect in each year of the timeframe, if possible.

Columns 13-15: the integer numeric site ID (from File9.txt) of the storage site to which the indicated project is to be preallocated in the indicated year.

Columns 16-20: the integer numeric unit set ID (from File8.txt) of the unit set to be allocated by designation.

Column 21-25: an integer indicating the year the allocation of the indicated unit set is to take effect. This value must be between 1 and 8 or must be equal to 99. If the value, n, is between 1 and 8, the designated allocation will take effect, if possible, in year n. If the value is 99, the designated allocation will take effect in each year of the timeframe, if possible.

Columns 28-30: the integer numeric site ID (from File9.txt) of the storage site to which the indicated unit set is to be preallocated in the indicated year.

E-5. OPTIONAL MOE OUTPUT. In order to enable assessment of the SITING solution relative to problem Objectives 1 through 3, SITING can generate three optional output files, denoted as File14.txt, File17.txt, and File20.txt, to provide measures of effectiveness on how well the solution plan meets these objectives. The generation of these optional output files depends upon user specifications set in columns 19-20 of optional input File15.txt. An informational measure is also computed. The SITING MOEs are summarized in Table E-2 according to the SITING objective(s) they treat and are cross-referenced with the field in the associated SITING output file which contains that information. These MOEs are described in paragraph E-6 below.

Table E-2. SITING Solution Assessment MOEs

MOE	What MOE measures	File
Avg tons x km for unit set moved from storage site to GDP	Objective 1 (nearness to GDP)	File17.txt
Fraction of unit sets changing storage site from previous yr	Objective 2 (resiting turbulence)	File17.txt
Number of sites over which each project is stored	Objective 3 (project dispersion)	File17.txt
Largest fraction of each project stored at a single site	Objective 3 (project dispersion)	File17.txt
Total number of sites used to store all projects	Objective 3 (project dispersion)	File17.txt
Fraction capacity filled at each storage site	Storage site utilization	File14.txt

E-6. MOE DESCRIPTION

a. MOE for Objective 1 (unit set positioning). Average weight-distance (tons x km) per unit set reflected in the movement requirement from the solution storage site for a unit set to the unit set GDP location. This is akin to ton-mile movement requirements in logistics. This MOE is computed as an average over the unit sets in each project in each year (column 21-30 of optional output File17.txt). An overall average over all unit sets is also done for each year (columns 21-30 of optional output File17.txt). If the user elects to generate File 20, this MOE is given for each individual unit set allocated (columns 43-52 of optional output File20.txt).

b. MOE for Objective 2 (Turbulence). The fraction of unit sets which had to be resited in each year, i.e., the fraction of unit sets in each year whose solution storage site is different from that unit's solution site in the previous year. (For the first year, a unit set's in-place site is treated as its previous year's storage site.) This MOE is computed for each individual project in each year as well as over all unit sets (columns 51-55 in optional output File17.txt).

c. MOEs for Objective 3 (project dispersion)

(1) The number of different storage sites over which each project is stored in the SITING solution (columns 31-40 of optional output File17.txt).

(2) The average number of different storage sites over which an average project is stored in the SITING solution (columns 31-40 of optional output File17.txt).

(3) The decimal fraction of the project represented by the largest "piece" (cluster of allocated unit sets) from the project which is stored at a single site (columns 11-20 of optional output File17.txt). This is a measure of project integrity in storage. The closer this MOE is to 1.00, the closer the project is to being stored at a single site.

(4) The total number of storage sites used (i.e., with at least one unit set allocated) to store all the unit sets in the solution (columns 31-40 of output optional File17.txt).

d. Informational Measures. Several SITING informational measures are optionally written onto output files denoted as File14.txt, File18.txt, and File21.txt. These measures are summarized in Table E-3 and are also described below. The informational measures described are also cross-referenced with the field in the associated SITING output file which contains that information.

Table E-3. SITING Solution Informational Measures

Informational measure	What is measured	Output file
Fraction capacity filled at each storage site	Storage site utilization efficiency	File14.txt
For each unit set, the preference rank of each site relative to distance from unit set GDP	Objective 1 (positioning)	File21.txt
For each project, the preference rank of each site relative to distance from the avg GDP of units in the project	Objective 1 (positioning)	File18.txt

(1) The fraction of capacity in each storage site which is filled by solution allocations (of unit sets) in each year (columns 18-22 in optional output file14.txt).

(2) A unit set site preference list (output file21.txt) showing for each unit set in the final year showing, for each unit set, the rank order of storage sites according to their meeting Objective 1 (closeness to unit set GDP). The list also shows the distance (columns 12-21 in output File21.txt) between each unit set-site combination in the list. The user can use this list to assess whether a solution allocation has picked the best, second best, etc site to store a unit set, relative to Objective 1. This list is useful as a general guide for allocation of unit sets.

(3) A project site preference list (output File18.txt) showing, for each project in a year, the rank order of storage sites according to their meeting Objective 1 for the entire group of unit sets comprising that project. The list also shows the average distance (columns 13-22) between an average unit set in the project and each storage site. The user can use this list to assess whether a solution allocating an entire project to a site has picked the best, second best, etc., site to store the project, relative to Objective 1.

E-7. **OVERVIEW OF OPTIONAL OUTPUT.** A comprehensive overview of the items described in SITING output is shown in Table E-4. The "Item" categories and associated characteristics are:

a. Projects. A characteristic described here characterizes the scenario attributes of individual projects.

b. Unit Sets. A characteristic described here characterizes the scenario attributes of individual unit sets.

c. **Individual Unit Set Allocations.** A characteristic described here characterizes an attribute of the SITING solution allocation for a specific unit set.

d. **Individual Project Allocations.** A characteristic described here characterizes an attribute of the SITING solution allocation for (all of the unit sets in) a specific project.

e. **All Allocations.** A characteristic described here characterizes an attribute of the overall SITING solution.

f. **Miscellaneous.** All attributes not characterized by paragraphs E-7a through e above.

The SITING output file containing the item characteristic is noted in the column headed "output file" in the table.

Table E-4. Items and Characteristics Described in Optional SITING Outputs

Item	Characteristic described	Output file
Projects	For each project, the closeness of each storage site to the "project GDP"	File18.txt
	For each project, the ranked list of storage sites according to "closeness to "project GDP"	File18.txt
Unit sets	For each unit set, the closeness of each storage site to the unit set's GDP	File21.txt
	For each unit set, the ranked list of storage sites according to closeness to unit set GDP	File21.txt
	The closest site to each unit set's GDP	File20.txt
	For each unit set, the site to which it was allocated in the previous yr	File20.txt
	Size of each unit set	File20.txt
	Weight of each unit set	File20.txt
Individual unit set allocations	Allocation site assigned each unit set	File20.txt
	(weight of unit)x(distance from unit's allocation site to its GDP)	File20.txt
	(weight of unit)X(distance from allocation site to allocation site in previous yr)	File20.txt
Individual project allocations	Average of [(weight of unit)x(distance fromunit's allocation site to its GDP)] over all units in project	File17.txt
	Number of sites used to store each project	File17.txt
	Largest fraction of project allocated to a single site	File17.txt
	Fraction of unit sets in project changing storage site from previous yr	File17.txt
All allocations	Average of [(weight of unit)x(distance from unit's allocation site to its GDP)] over all units allocated	File17.txt
	Total number of nonempty storage sites	File17.txt
	Fraction of all allocated unit sets changing storage site from previous yr	File17.txt
	Total amount (size) of unit sets allocated to each site	File14.txt
	Fraction of each storage site's capacity occupied by allocations	File14.txt
Miscellaneous	Distance between each pairing of storage sites	File16.txt

E-8. STRUCTURE OF OPTIONAL OUTPUT FILES. The following defines the structure of all optional formatted output files which can be generated by SITING through use of input File15.txt. These comprise File14.txt, File17.txt, File18.txt, and File20.txt. In addition, the structure of the modified output File16.txt (as specified in input File15.txt) is also given. The descriptive names of these optional output files are shown in Table E-5. When the user has used input File15.txt to specify the writing of column headings in output files, the headings used are given in brackets [ ] at the beginning of the associated column field definition given below.

Table E-5. Optional Formatted SITING Output Files

File	Descriptive name
File14.txt	Site fill report
File16.txt	Modified site-site distance listing
File17.txt	MOE summary
File18.txt	Project site preference lists
File20.txt	Detailed siting characteristics list

a. Site Fill Report. Output File14.txt is the site fill report which is created in the allocation phase of SITING. It summarizes the site fill (fraction of storage site capacity used by allocations) over the storage sites after allocations are done. Output column headings are as indicated in brackets. Its items are:

(1) Columns 1-5: [YR] The integer year being processed (1 through 8).

(2) Columns 6-10: [SITE] The integer numeric ID of the storage site being reported

(3) Columns 11-17: [AMOUNT STORED] The integer total area occupied (allocated) by all unit sets allocated to that site in that year.

(4) Columns 18-22: [FRAC FILL] The decimal fraction fill, defined here as the ratio of total site area occupied to total site capacity.

Note that the last two sites reported on are site -01 and site -99. These are fictitious sites which are model constructs. Site -99 is an overflow site which stores units only when there is no room to allocate them at any "actual" site. If this site becomes full (has at least 200 unit sets allocated to it), then the remaining overflow goes to site -01. If there is no overflow, these two sites will always be empty. If SITING generates any overflow allocations to these two sites, then the allocation problem is not really solvable because there is not enough storage capacity available to

store all of the unit sets. This makes the SITING solution literally invalid, since the overflow was not really allocated to an actual site.

b. Modified Site-Site Distance Listing Output File. This consists of output File16.txt when the user specifies its generation in input File15.txt. In such a case, the format of File16.txt is different from the default (without File15.txt) generation. (Output column headings are as shown in brackets.)

(1) Columns 1-5: [SITE] The integer numeric site ID of first member of site pair.

(2) Columns 6-10: [SITE] The integer numeric site ID of second member of site pair.

(3) Columns 11-19: [DIST(KM)] A decimal equal to the distance (in km) between members of this site pair]).

c. MOE Summary Output File. File17.txt is the MOE summary created by the SITING assessment phase. The SITING default is to omit File17.txt. It shows measures of effectiveness describing how well the SITING solution siting plan meets the methodology objectives. For each year it consists of one line for each project summary and one total (i.e., all projects) summary line for the year. The total summary line has the same format structure, but is marked by a " AV=" entry in columns 1-4 (which are blank for the individual project summary lines). The items are:

(1) Individual Project Summaries

(a) Columns 5-7: [YR] The integer year being processed (1 through 8).

(b) Columns 8-11: [PRJ] The integer project ID of the project being summarized.

(c) Columns 12-21: [MAX FRAC AT 1 SITE] The decimal fraction of the project represented by the largest "piece" (cluster of allocated unit sets) from the project which is stored at a single site. This is a measure of project integrity and reflects on how well Objective 3 is met. The closer this entry is to 1.00, the closer the project is to being stored at a single site.

(d) Columns 22-31: [AVG UNIT KM TO GDP] The decimal arithmetic average of [unit set weight x distance from allocation site to unit GDP] over all allocated units of this project, excluding unit sets allocated to the overflow sites. This is a measure of how well Objective 1 is met.

(e) Columns 32-41: [NR SITES (AV=AVG)] A decimal number equal to the number of storage sites ("pieces") over which the allocated units of the project are dispersed. The closer to 1.00 this is, the better Objective 3 is met.

(f) Columns 42-51: [(AV=TOT)] The item (e) is repeated here.

(g) Columns 52-56: [FRAC MOVE] The decimal fraction of the allocated unit sets in this year which have been resited from their allocation sites in the previous year. An allocated unit set is said to be resited if it was present in the previous year and has a different allocation site this year from the allocation site it had in the previous year. For the first year, resiting is relative to the input in-place siting (from File11.txt). This is a measure of Objective 2 (keep resiting turbulence small).

(h) Columns 57-65: [TOT UNITS ALLOC] The integer number of allocated unit sets in this project.

(2) Total (All Projects) Summary

(a) Columns 1-4: A blank followed by the characters AV= identify this line as a total summary line.

(b) Columns 4-7: [YR] The integer year being processed.

(c) Columns 22-31: [AVG UNIT KM TO GDP] The decimal arithmetic average of [unit set weight x distance from allocation site to unit GDP] over all allocated units, excluding unit sets allocated to the overflow sites. This is a measure of how well Objective 1 is met.

(d) Columns 32-41: [NR SITES (AV=AVG)] A decimal number equal to the average number of storage sites (pieces) over which the allocated units of an average project are dispersed. This is just the arithmetic average of the individual project measures listed in the individual project summaries. The closer to 1.00 this is, the better Objective 3 is met.

(e) Columns 42-51: [(AV=TOT)] The decimal number of storage sites over which all allocated units for the year are dispersed. This is a measure of how well Objective 3 is met.

(f) Columns 52-56: [FRAC MOVE] The decimal fraction of the allocated unit sets in this year which have been resited from their allocation sites in the previous year. An allocated unit set is said to be resited if it was present in the previous year and has a different allocation site this year from the allocation site it had in the previous year. For the first year, resiting is relative to the input in-place siting (from File11.txt).

(g) Columns 57-65: [TOT UNITS ALLOC] The integer total number of unit sets allocated this year.

d. Project Site Preference List Output File. File18.txt consists of the project site preference lists. For each project processed in a year, this file gives the rank ordering of all storage sites relative to the average distance (per ton) from that storage site to the project GDP for that project. The project GDP is analogous to a centroid location of all the unit sets in the project. The associated distance is also given. This file consists of a series of such site preference lists for all unit sets processed in one or more years. The order of the lists is the order of the unit sets in the input File8.txt for that year. Each record in File18.txt

gives site preference information for a specific combination of year, unit set, and site. The structure of each record is:

(1) Columns 1-5: [UNIT] The integer numeric site ID for this record.

(2) Columns 13-22: [KM TO GDP] A decimal number equal to the average distance (in km) between the indicated project's project GDP and the indicated site.

(3) Columns 23-32: [PROJ] The integer numeric project ID for this record.

(4) Columns 23-37: [YR] The year for which this record is applicable (1 through 9, where 9 is equivalent to the goal year).

e. Detailed Siting Characteristics List Output File. File20.txt is the detailed siting characteristics list. It consists essentially of the items in File12.txt, with some extra descriptors, but, for each year, these are sorted in a different way from File12.txt. The first sort is by project. Each record reports on a single unit set allocation. Within projects, items are sorted by site. This is a very useful sorting for analysis. The items within this file are:

(1) Columns 1-3: [YR] The integer year being processed.

(2) Columns 4-7: [PRJ] The integer numeric project ID to which the allocated unit belongs.

(3) Columns 8-11: [UNT] The integer numeric unit set ID of the allocated unit.

(4) Columns 12-15: [ALL SIT] The integer numeric ID of the site to which this unit set is allocated.

(5) Columns 16-23: [UNIT AREA] The integer total storage area (from input File8.txt) in this allocated unit set.

(6) Columns 24-31: [UNIT TONS] The integer total weight (from input File8.txt) of this allocated unit set.

(7) Columns 32-35: [IP SIT] The integer numeric ID of the site at which this allocated unit set was allocated in the previous year. (For year 1, this is the in-place site from input File11.txt.)

(8) Columns 36-45: [TONS X KM RESITED] A decimal equal to (weight of allocated unit set) x (distance from the unit set's allocation site this year and its allocation site in the previous year).

(9) Columns 46-55: [TONS X KM TO GDP] A decimal equal to (weight of allocated unit set) x (distance from the unit set's allocation site to the unit set's GDP)

(10) Columns 56-59: [OPT SITE] The integer numeric ID of the site closest to the GDP of this allocated unit set.

APPENDIX F

SITING SOURCE CODE DOCUMENTATION

F-1. INTRODUCTION. This appendix contains the source code documentation for the SITING model. Primarily written in FORTRAN, SITING code is included in this appendix as a reference tool for system users. The user is cautioned that while the displayed source code represents the SITING Model at the time of publication, it is not a final and unique form. SITING must be regarded as a dynamic model which is continually being refined and improved.

F-2. SITING DOCUMENTATION

```
$LARGE
$INCLUDE:'COMMON.INC'
$INCLUDE:'NEWCOM.INC'
C
C  *** SITER = ++ SEQUENTIAL IMPROVEMENT TECHNIQUE FOR EFFECTIVE RESITING ++
C  *** DESIGNED BY    WALTER J. BAUMAN
C                     USACAA (ATTN: CSCA-FSC)
C                     8120 WOODMONT AVE
C                     BETHESDA, MD 20814-2797
C
C  SITER WAS DEVELOPED PRIMARILY FOR USE IN THE CAA POMCUSITES STUDY.
C  ITS PURPOSE IN THAT STUDY WAS TO ASSIST POMCUS STORAGE PLANNERS
C  IN THE SITING OF POMCUS UNIT SETS WITH THE FOLLOWING OBJECTIVES :
C
C      1. STORE UNIT SETS CLOSE TO ASSOCIATED UNIT GDP ASSEMBLY AREAS.
C      2. REDUCE THE AMOUNT OF RESITING (CHANGE IN SITE) OF STORED
C         UNIT SETS.
C      3. (OPTIONALLY) REDUCE THE NUMBER OF SITES REQUIRED TO
C         STORE ALL THE UNIT SETS COMPRISING A PROJECT CLUSTER.
C
C
C
      DIMENSION
     +    DLAT(MXSITE),
     +    DLON(MXSITE),        IIPUN(MXUNIT),
     +    IIU(MXUNIT),         INPRI(MXUNIT),       ISPR(MXSITE),
     +    ITRANS(MXSITE),      IWW(MXSITE),         KTRANS(MXUNIT),
     +    LLABSI(MXSITE)
      REAL  ISUM, IWW, JWT
      CHARACTER*3
     +    LAB,      LLABSI,   SIP
      CHARACTER*1
     +    LOC

      ISK = 0
  776 ISTOP=0
      IERR=0
C
C  SET DEFAULT VALUES OF VARIABLES.
C
      IND5=0
      IPR12 =1
      IPR16 = 0
      IPR17 = 0
      IPR18 = 0
      IPR19 = 0
      IPR20 =0
      IBUGYR = 0
      ICYR = 9
      IHD = 0
      IF (ISK .EQ. 0)THEN
      CALL SET (NSLIM, IND1, IPAS)
```

```
          DO 12 M = 1,8
          DO 13 L = 1,MXPKG
   13     ZNUC(M,L)=0
   12     CONTINUE
        ENDIF
        OPEN (7,FILE='FILE7.txt')
        OPEN (8,FILE='FILE8.txt')
        OPEN (9,FILE='FILE9.txt')
        OPEN (10,FILE='FILE10.txt')
        OPEN (11,FILE='FILE11.txt')
        OPEN (15,FILE='FILE15.txt')
        OPEN (12,FILE='FILE12.txt')
        OPEN (14,FILE='FILE14.txt')
        OPEN (16,FILE='FILE16.txt')
        OPEN (17,FILE='FILE17.txt')
        OPEN (18,FILE='FILE18.txt')
        OPEN (20,FILE='FILE20.txt')
        OPEN (21,FILE='FILE21.txt')
C
C      IF DESIRED, OVERRIDE DEFAULTS FOR OPERATIONAL PARAMETERS BY
C      READING MODIFIED OPERATIONAL PARAMETERS FROM FILE15.
C
          READ(15,24,ERR = 6,END = 6) NFIN,IPR12,IPR16,IPR17,IPR18,
     +    IPR19,IPR20,ICYR,IHD,IBUGYR
          IF (IPAS .EQ. 2)NSLIM = 99
   24   FORMAT(10(3x,I2))
C
        IF (IHD .NE. 0 .AND. IPR17 .NE. 0)THEN
          WRITE(14,31)
   31 FORMAT(10X,' AMOUNT FRAC',/3X,'YR SITE STORED FILL')
          IF (ISK .EQ. 0)THEN
            IF(IPR17 .NE. 0)WRITE(17,32)
   32 FORMAT(13X,'MAX FRAC',2X,'AVG WT X  NR SITES',10X,' FRAC TOT ',
     +'UNITS',/,5X,'YR PRJ AT 1 SITE KM TO GDP  (AV=AVG)  (AV=TOT) ',
     +'MOVE',4X,'ALLOC')
            IF(IPR20 .NE. 0)WRITE(20,33)
   33 FORMAT(12X,'ALL    UNIT    UNIT  IP  TON X KM  TON X KM OPT',
     +/,1x,'YR PRJ UNT SIT    AREA    TONS SIT   RESITED    TO GDP',
     +' SITE')
          ENDIF
        ENDIF
C
C    BRANCH IF THIS IS THE ASESSMENT PHASE OR IF THIS IS A
C    BASELINE CASE (IN-PLACE PROCESSING ONLY)
C
   17   IF (ISK .EQ. 1) GO TO 6
        IF (IPAS .EQ. 3)GO TO 53
        WRITE(*,23)
        IF (IPR19 .NE. 0)WRITE(19,23)
   23   FORMAT(/,1X,' *** FILE15 IS INPUT IN THIS RUN ***',/)
        WRITE(*,7)NFIN
        IF (IPR19 .NE. 0)WRITE(19,7)NFIN
    7   FORMAT(1X,'  NR YEARS IN',
     +' TIMEFRAME =',I3)
```

```
C
C  SET DEFAULT VALUES FOR ASSESSMENT PHASE
C
    6  IF (ISK .EQ. 1)  THEN
         IND5=1
         ISTOP=1
         WRITE (*,10)
   10    FORMAT (//,1X,' ** SITER ASSESSMENT PHASE BEGINS **')
C
C  COUNT THE NUMBER OF RECORDS(ALLOCATIONS) FOR EACH YEAR IN INPUT
C  FILE 12 DURING ASSESSMENT PHASE.
C
         DO 20 I=1,9
   20    NCNT(I)=0
   30    READ (12,40,END=50) LYR
   40    FORMAT (I3)
         NCNT(LYR)=NCNT(LYR)+1
         GO TO 30
   50    REWIND 12
       ENDIF
C
C  SET DEFAULT INPUTS FOR BASELINE CASE (IN-PLACE ASSESSMENT)
C
   53 IF (IPAS .EQ. 3)THEN
         NFIN = 1
         IND5 = -1
         ISTOP = 1
         WRITE(*,55)
         IF(IPR19 .NE. 0)WRITE(19,55)
   55    FORMAT (/,1X,' ** BEGIN ASSESSMENT OF IN-PLACE IN YEAR 1 **',/)
       ENDIF
C
C  IF THIS RUN IS IN AN ALLOCATION PHASE, .I.E. THIS IS NOT AN  ASSESSMENT
C  PHASE OR A BASELINE CASE (IN-PLACE ASSESSMENT ONLY), ADD 1 TO NFIN TO
C  GET TOTAL YRS RUN (SINCE THE GOAL (FINAL) YEAR IS RUN TWICE - AT START
C  AND AT END).
C
       IF (IND5.EQ.0) NFIN=NFIN+1
       IF (IND5.EQ.0) WRITE (*,60)
   60 FORMAT (//,1X,' ** SITER ALLOCATIONS BEGIN **')
C
C  IF THIS RUN IS IN ALLOCATION PHASE AND IS NOT IN ASSESSMENT PHASE
C  OR IN BASELINE CASE (IN-PLACE ASSESSMENT ONLY), THEN INITIALLY
C  POSITION FILE8, FILE9, AND FILE10 AT THE BEGINNING OF DATA FOR
C  THE GOAL (FINAL) YEAR.
C
       IF (ISTOP.EQ.0)  THEN
         INC=1
   80    READ (8,90,END=100) NG
   90    FORMAT (16I5)
         IF (NG.LT.0)  THEN
           INC=INC+1
           IF (INC.EQ.(NFIN-1)) GO TO 100
         ENDIF
```

```
        GO TO 80
  100   INC=1
  110   READ (9,120,END=130) lab
  120   FORMAT (2X,A3,I10)
        IF (lab.eq.'999')  THEN
          INC=INC+1
          IF (INC.EQ.(NFIN-1)) GO TO 130
        ENDIF
        GO TO 110
  130   INC=1
  140   READ (10,90,END=150) NG
        IF (NG.GE.9999)  THEN
          INC=INC+1
          IF (INC.EQ.NFIN-1) GO TO 150
        ENDIF
        GO TO 140
      ENDIF
C
C  INITIALIZE ARRAYS AND WRITE HEADINGS
C
  150 DO 160 K=1,MXHOLD
        LMUREQ(K)=0
  160 IHOLD(K)=0
      DO 165 N = 1,MXPKG
      ICSTAY(N) = 0
  165 IPSTAY(N) = 0
      DO 170 K=1,MXUNIT
      MUSIT(K) = 0
      LLMURQ(K) = 0
  170 IIU(K)=0
C
C  READ IN (IF ANY) PROJECTS AND UNITS TO BE ALLOCATED TO DESIGNATED SITES
C  IN DESIGNATED YEARS BEFORE OPERATION OF THE NORMAL SITER ALLOCATOION
C  ALGORITHM.
C
      IF (IPAS .EQ. 1)THEN
        NN=1
  190   READ (7,200,ERR=220,END=220)IFPN(NN),IFPY(NN),IFPS(NN),IFUN(NN),
     +IFUY(NN),IFUS(NN)
  200   FORMAT (2I5,2X,A3,2I5,2X,A3)
        IF (NN .EQ. 1) WRITE (*,191)
  191   FORMAT (/,1X,' ** FILE7 IS INPUT IN THIS RUN **',/)
        NN=NN+1
C
C  TRUNCATE FILE 7 INPUT IF IT HAS MORE THAN MXFL RECORDS.
C
        IF (NN.GT.MXFL)  THEN
          IF(IPR19 .NE. 0)WRITE (19,210) MXFL
  210   FORMAT (1X,' DESIGNATED ALLOCATION LIST TRUNCATED -',' EXCEEDS',
     +  I5,' ITEMS')
          GO TO 220
        ENDIF
        GO TO 190
      ENDIF
```

```
  220 MFL=NN-1
      REWIND 7
C
C ** LOOP PROCESSING YEAR BY YEAR
C
      DO 1240 NYR=1,NFIN
        IDUN=0
        IDUNP=0
        JFUL = 0
C  FOR EACH YEAR AFTER THE FIRST YEAR, SET LMSIT, LNSIT, AND LLABSI( )
C  EQUAL TO TO THE PREVIOUS YEAR'S VALUES OF MSIT, NSIT, AND LABSI( )
C  RESPECTIVELY.
C
        IF (NYR.GT.1)  THEN
          LMSIT=MSIT
          LNSIT=NSIT
          DO 230 I=1,MSIT
  230     LLABSI(I)=LABSI(I)
          LNTUN=NTUN
          DO 240 K=1,LNTUN
            IIPUN(K)=IPUN(K)
            INPRI(K)=NPRI(K)
  240     IIU(K)=IU(K)
        ENDIF
C
C  INITIALIZE ARRAYS, ETC FOR THIS YEAR
C
        DO 250 K=1,MXUNIT
          ISET(K)=0
          IU(K)=0
          KTRANS(K)=0
          IPUN(K)=0
          JUN(K) = 0
          MOF(K)=0
          JPRI(K)=0
          NPRI(K)=0
  250   JSIT(K)='???'
        MPCT=0
        NB=0
        IMTOT=0
        IMALL = 0
        XDG=0
        MOEG=0
        DO 270 I=1,MXPKG
          DO 260 L=1,MXLIST
  260     IDPKG(I,L)=0
          NUCP(I) = 0
          MPREQ(I)=0
          JKP(I)=0
  270     IPRIO(I)=-998
        DO 290 NS=1,MXSITE
          DO 275 MS = 1,MXSITE
  275     JDIST(NS,MS) = 99999
          DO 280 L=1,MXLIST
```

```
  280     INPLU(L,NS)=0
          NUIS(NS)=0
          NSCAP(NS)=0
          IRES(NS)=0
          IFUL(NS) = 0
  290   CONTINUE
C
C  START READING FILE 8 (UNIT SET DATA) FOR THIS YEAR
C
        READ (8,300) MYR
  300   FORMAT (1X,I2)
        IF (IBUGYR .NE. 0) WRITE (19,310) MYR
  310   FORMAT (//,1X,'------------  START OF YEAR',I3,'  -------------
     +--------',//////)
C
C  START READING FILE 9 (SITE DATA) FOR THIS YEAR. IF YEAR DOES
C  NOT AGREE WITH THAT FROM FILE 8, WRITE WARNING ON FILE 19.
C
        READ (9,300) IYR
        IF (IYR.NE.MYR) WRITE (*,320) IYR,MYR
  320   FORMAT (' ** WARNING: FILE 9- YEAR',I3,'  IS NOT =',I3,' FROM F
     + ILE 8')
        IF (IYR.NE.MYR .AND. IPR19 .NE. 0) WRITE (19,320) IYR,MYR
        NN=1
C
C  FOR EACH SITE, READ (FROM FILE 9) LABEL ID, CAPACITY, AND LOCATION
C  (LATITUDE,LONGITUDE).  CONVERT LAT,LONG TO RADIANS.
C
        DO 340 N=1,MXSITE
          READ (9,330,END=350) LAB,MSCAP,LAUD,LAUM,LAUS,LOUD,LOUM,LOUS,L
     +    OC
  330     FORMAT (2X,A3,I10,I5,2I2,1X,I5,2I2,A1,I5)
          IF (LAB.EQ.'999') GO TO 350
          IF (LAUD.GT.0)  THEN
C
C  IF THIS IS UNCONSTRAINED STORAGE CASE, SET EACH SITE CAPACITY
C  TO A VERY LARGE NUMBER
C  INITIALIZE SITE RETENTION PRIORITY TO -(SITE CAPACITY) SO THAT
C  SMALLEST SITES HAVE LEAST RETAINABILITY
C
            IF(IPAS .EQ. 2)MSCAP = 99999999
            ISPR(NN)=-MSCAP
            LABSI(NN)=LAB
            NSCAP(NN)=MSCAP
            DLAT(NN)=LAUD+FLOAT(LAUM)/60.+FLOAT(LAUS)/3600.
            DLON(NN)=LOUD+FLOAT(LOUM)/60.+FLOAT(LOUS)/3600.
            IF (LOC.EQ.'W') DLON(NN)=360.-DLON(NN)
            DLAT(NN)=DLAT(NN)*.01745329
            DLON(NN)=DLON(NN)*.01745329
            NN=NN+1
          ENDIF
  340   CONTINUE
  350   NSIT=NN
C
```

```
C  IF A SITE IS NEW THIS YEAR, ADD IT AND INDEX IT TO THE INPUT SITE LIST
C
          DO 351 I=1,LNSIT-1
             IH = 0
             DO 352 NS=1,NSIT-1
               IF (LABSI(NS).EQ.LLABSI(I))  THEN
                 IH = 1
                 GO TO 351
               ENDIF
  352        CONTINUE
             IF (IH.EQ.0)  THEN
               ISPR(NSIT) = 0
               LABSI(NSIT) = LLABSI(I)
               NSCAP(NSIT) = 0
               DLAT(NSIT) = 0
               DLON(NSIT) = 0
               NSIT = NSIT + 1
             ENDIF
  351     CONTINUE
C
C  NOTE ERROR IF TOTAL NR OF INPUT SITES EXCEEDS (MXSITE).
C
        IF (NSIT.GT.(MXSITE))  THEN
          IERR=1
          WRITE (*,360) NSIT,MYR,MXSITE
  360     FORMAT (1X,'FATAL ERROR: NR SITES=',I4,' FROM FILE 9 FOR YEAR'
     +,I4,' EXCEEDS',I4)
          IF(IPR19 .NE. 0)WRITE (19,360) NSIT,MYR,MXSITE-1
          NSIT=MXSITE-1
        ENDIF
        LSIT=NSIT
        MSIT=NN+1
C
C  LABEL LAST SITE BLANK AS A PLACEHOLDER .
C  LABEL PHANTOM 'CONUS SITE' (LSIT) as '-01'.
C  LABEL PHANTOM 'OVERFLOW SITE' (MSIT) as '-99'.
C
        LABSI(MXSITE)='   '
        LABSI(LSIT)='-01'
        LABSI(MSIT)='-99'
C
C  CONSTRUCT TABLE OF DISTANCES (KM) BETWEEN SITES
C
        DO 380 NS=1,NSIT
          DO 370 MS=1,NSIT
             IF (DLAT(NS).LE.0.OR.DLAT(MS).LE.0)  THEN
               JDIST(NS,MS)=0
             ELSE
               DLAU1=DLAT(NS)
               DLOU1=DLON(NS)
               DLAU=DLAT(MS)
               DLOU=DLON(MS)
               JDIST(NS,MS) = 10.*( CONVRT(DLAU1,DLOU1,DLAU,DLOU) +.05 )
             ENDIF
```

```
  370       CONTINUE
  380     CONTINUE
          DO 390 K=1,MXUNIT
            IDIST(NSIT,K) = 0
            IDIST(NSIT+1,K) = 0
            IUSIT(K)=0
            MUWT(K)=0
            IMIN(K)=MXSITE
            NPRI(K)=0
  390     MUREQ(K)=0
          LTOT=0
          LITOT=0
          MALL=0
C
C  READ, FROM FILE 8, UNIT SET CHARACTERISTICS (UNIT NR, SIZE,
C  WEIGHT, LOCATION OF ASSOCIATED GDP, AND PRIORITY.
C  CONVERT UNIT SET GDP LOCATION FROM LAT,LONG TO RADIANS
C
          N=0
          IMAXPR=-9999
  400     READ (8,410,END=470) IN,MUR,MUW,LAUD,LAUM,LAUS,LOUD,LOUM,LOUS,LO
     +    C,NPR
  410     FORMAT (I5,2I10,I5,2I2,1X,I5,2I2,A1,I7)
          IF (IN.LT.0) GO TO 470
C
C  NOTE ERROR IF AN INPUT UNIT SET ID EXCEEDS MXHOLD.
C
          IF (IN.GT.MXHOLD)  THEN
            WRITE (*,420) IN,MYR,MXHOLD
  420       FORMAT (1X,'** FATAL ERROR : UNIT',I5,' IN YEAR',I3,' HAS UNIT
     + SET ID EXCEEDING',I5)
            IERR=1
          ENDIF
C
C  WRITE NOTICE THAT A UNIT SET WITH INPUT SIZE = WT = 0 IS BEING IGNORED.

C
          IF (MUR.LE.0.AND.MUW.LE.0) THEN
            IF(IND5.EQ. 0)WRITE (*,430) IN,MYR
  430       FORMAT (' UNIT SET',I5,' IN YR',I3,' IGNORED: SIZE = WT = 0')
            IF(IND5.EQ. 0 .AND. IPR19 .NE. 0)WRITE (19,430) IN,MYR
            GO TO 400
          ENDIF
C
C  WRITE NOTICE THAT AN INPUT UNIT SET AT ZERO DEGREES LATITUDE IS
C  BEING IGNORED (BECAUSE PROBABLY NO LAT, LONG LOCATION IS INPUT).
C
           IF (LAUD.LE.0)  THEN
            IF(IND5.EQ. 0)WRITE (*,440) IN,MYR
  440       FORMAT (1X,' UNIT SET ',I5,' IN YR',I3,' IGNORED -',' NO',
     +' LAT,LONG LOCATION')
            IF(IND5.EQ.0.AND. IPR19 .NE. 0)WRITE (19,440) IN,MYR
            GO TO 400
          ENDIF
```

```
      LOC='E'
      N=N+1
      IU(N)=IN
      MUREQ(N)=MUR
      MUWT(N)=MUW
      JPRI(N)=NPR
      NPRI(N)=NPR
      IMAXPR=MAX(NPR,IMAXPR)
      IF (LAUD.GT.0)  THEN
        LITOT=LITOT+1
        MALL=MALL+MUREQ(N)
        LTOT=LTOT+MUWT(N)
        ULAT=LAUD+FLOAT(LAUM)/60.+FLOAT(LAUS)/3600.
        ULON=LOUD+FLOAT(LOUM)/60.+FLOAT(LOUS)/3600.
        IF (LOC.EQ.'W') ULON=360.-ULON
        ULAT=ULAT*.01745329
        ULON=ULON*.01745329
      ENDIF
      LDIS(N)=99999999
C
C  CONSTRUCT TABLE OF DISTANCES(KM) BETWEEN SITES AND UNIT GDP LOCATIONS.
C  FOR EACH UNIT N, DETERMINE SITE CLOSEST TO UNIT GDP (IMIN(N) AND
C  THE [SITE-UNIT GDP] DISTANCE (LDIS(N)) IN THIS (CLOSEST) CASE.
C
      DO 460 NS=1,NSIT-1
        IDIST(NS,N)=99999999
        IF (LAUD.GT.0)  THEN
          DLAU1=DLAT(NS)
          DLOU1=DLON(NS)
          DLAU=ULAT
          DLOU=ULON
          IDIST(NS,N)= 10.*( CONVRT(DLAU1,DLOU1,DLAU,DLOU) + .05)
          IF (IDIST(NS,N).LT.LDIS(N).AND.NSCAP(NS).GT.0)  THEN
            IMIN(N)=NS
            LDIS(N)=IDIST(NS,N)
          ENDIF
        ENDIF
 460  CONTINUE
      GO TO 400
C
C  SEEK MSEL = LARGEST POWER OF 10 WHICH EXCEEDS THE LARGEST INPUT
C  UNIT PRIORITY.
C
 470  NTUN=N
      MNUN = NTUN
        IF(NTUN .GT. MXUNIT)THEN
          IERR = 1
          WRITE(*,471)NTUN,MYR,MXUNIT
          IF (IPR19 .NE. 0)WRITE(19,471)NTUN,MYR,MXUNIT
 471      FORMAT(1X,' FATAL ERROR:',I6,' = NR OF UNITS IN YR',I3,

    +     ' EXCEEDS',I6)
        ENDIF
      DO 480 NP=1,10
```

```
         Z = 10.**NP
         IF (Z.GT.FLOAT(IMAXPR))THEN
           MSEL=NP
           GO TO 490
         ENDIF
  480   CONTINUE
C
C  START READING FILE 10 (PROJECT COMPOSITION) INPUT DATA FOR YEAR.
C
  490   READ (10,300) IYR
        IF (IYR.EQ.999) GO TO 470
        IF (IYR.NE.MYR) WRITE (*,500) IYR,MYR
  500   FORMAT ('** WARNING: FILE 10- YEAR',I3,'  IS NOT =',I3,' FROM
     +FILE 8')
        IF (IYR.NE.MYR .AND. IPR19 .NE. 0) WRITE (19,500) IYR,MYR
        IF (IYR.GT.8.OR.IYR.LT.0) GO TO 1250
        DO 510 K=1,MXPKG
          IPRIO(K)=0
          IX(K)=0
          INDS(K)=0
  510   NUNP(K)=0
        ZPKG= .00
        DO 620 I=1,MXPKG
C
C  READ PROJECT ID AND PRIORITY FOR EACH PROJECT IN FILE 10 THIS YEAR.
C
          READ (10,530,END=630) K,IPRI
  530     FORMAT (2I5,2F5.2,4I5)
          IF (K.LT.0) GO TO 620
          IF (K.GE.999) GO TO 630
C
C  NOTE ERROR IF AN INPUT PROJECT ID EXCEEDS MXPKG.
C
          IF (K.GT.MXPKG)  THEN
            IERR=1
            WRITE (*,540) K,MYR,MXPKG
  540       FORMAT (1X,'FATAL ERROR: PROJECT',I4,' IN FILE 10 FOR YEAR',
     +      I4,' HAS ID EXCEEDING',I6)
            IF(IPR19 .NE. 0)WRITE (19,540) K,MYR,MXPKG
            K=MXPKG
          ENDIF
          IPRIO(K)=IPRI
          NPAK=I
          MTOT(I)=0
C
C  READ, FROM FILE 10, UNIT SET ID (CROSS-REFERENCING THE INPUT UNIT SET
C  IDS FROM FILE 8) OF ALL UNITS IN EACH PACKAGE
C
          ID=0
          NN=0
  550     READ (10,90,END=600) IN
          IF (IN.LT.0) GO TO 600
          IF (IN.EQ.0) GO TO 550
C
```

```
C  NOTE ERROR IF ANY PROJECT HAS MORE THAN MXLIST UNITS IN IT.
C
          IF ((NN+1).GT.MXLIST)  THEN
            WRITE (*,560) K,MYR,MXLIST
  560       FORMAT (' FATAL ERROR: PROJECT',I3,' IN YR',I3,'HAS MORE',
     +' THAN', I4,' UNITS IN IT')
            IF (IPR19.NE. 0)WRITE (19,560) K,MYR,MXLIST
           IERR = 1
           GO TO 600
          ENDIF
          DO 570 KK=1,MXUNIT
            IF (IU(KK).EQ.IN)  THEN
              ID=KK
              GO TO 590
            ENDIF
  570     CONTINUE
          WRITE (*,580) IN,K,MYR
  580     FORMAT (1X,'WARNING: UNIT',I5,' IN PROJECT',I3,'FOR YEAR',
     +I3,' IS NOT DEFINED IN FILE8')
          IF(IPR19 .NE. 0)WRITE (19,580) IN,K,MYR
          GO TO 550
  590     IF (MUREQ(ID).LE.0.AND.MUWT(ID).LE.0) GO TO 550
          NN=NN+1
          IDPKG(K,NN)=ID
          MTOT(K)=MTOT(K)+1
          GO TO 550
  600     NUNP(K)=NN
C
C  CONSTRUCT INTERNAL UNIT PRIORITIES SUCH THAT, WHEN ALL UNIT SETS ARE
C  ORDERED BY INCREASING (INTERNAL) UNIT SET PRIORITY :
C
C  1.  ALL UNIT SETS WITHIN A PROJECT ARE TOGETHER, I.E. FORM A
C  'UNIT CLUSTER' IN THE ORDERED LIST.
C
C  2.  WITHIN THE UNIT CLUSTER FOR A PROJECT, ALL UNIT SETS ARE IN THE
C  SAME ORDER AS WOULD RESULT IF THE INPUT UNIT PRIORITIES (RATHER
C  THAN THE INTERNAL ONES) WERE THE BASIS FOR THE ORDERING.
C
C  3.  IF A PROJECT A HAS HIGHER PROJECT PRIORITY THAN A PROJECT B, THEN
C  THE UNIT CLUSTER FOR PROJECT A WILL PRECEDE THE UNIT CLUSTER FOR
C  PROJECT B IN THE ABOVE ORDERED LIST.
C
C  WITHIN THE ALLOCATION ALGORITHM, THE INTERNAL UNIT SET PRIORITIES WILL
C  BE USED IN PLACE OF THE INPUT UNIT SET PRIORITIES.
C
          IF (NN.EQ.0)  THEN
            IPRIO(K)=0
            GO TO 620
          ENDIF
          ZPKG=ZPKG+1.
          DO 610 N=1,NN
            IF (IDPKG(K,N).GT.0)  THEN
              NA=IDPKG(K,N)
              IPUN(NA)=K
```

```
            NPRI(NA)=(10.**MSEL)*IPRIO(K)+JPRI(NA)
            MPREQ(K)=MPREQ(K)+MUREQ(NA)
          ENDIF
  610   CONTINUE
        JKP(K)=MPREQ(K)
        IX(K)=IPRIO(K)
        INDS(K)=K
  620 CONTINUE
  630 CONTINUE
      DO 631 I = 1,NTUN
      IF ( (MUWT(I) + MUREQ(I)) .GT. 0
     + .AND. IPUN(I) .LE. 0)THEN
         IERR = 1
         WRITE (*,632)IU(I),MYR
         IF(IPR19 .NE. 0)WRITE(19,632)IU(I),MYR
  632   FORMAT(1X,' * FATAL ERROR: UNIT',I5, ' IN YR',I3,
     +  ' IS NOT ASSIGNED A PROJECT ID IN FILE10')
       ENDIF
  631  CONTINUE
C
C  ORDER PROJECT PRIORITIES IN ASCENDING SEQUENCE. STORE THAT
C  SEQUENCE IN ARRAY IPR.
C
      CALL ORDER (MXPKG)
      TMOV = 0
      DO 640 L=1,MXPKG
        IF (MYR .NE. 9 .AND. IND5*IPRIO(L) .NE. 0)
     +  TMOV =TMOV+ZNUC(MYR,L)*NUNP(L)
  640 IPR(L)=INDS(L)
C
C  IF THIS IS THE ASSESSMENT PHASE, CROSS-REFERENCE INPUT UNIT SET ID'S
C  FROM INPUT FILE 12 WITH UNIT SET ID'S FROM FILE 8.  ALSO ASSIGN EACH
C  UNIT SET FROM FILE 12 AS INPLACE AT THE ASSOCIATED SITE READ FROM FILE 12
C
      IF (IND5.GT.0)  THEN
        DO 650 I=1,NSIT
  650   NUIS(I)=0
        DO 700 I=1,NCNT(NYR)
          READ (12,660) LYR,JUN(I),JSIT(I),LV,LAB,SIP
  660     FORMAT (I3,I4,A3,I4,7X,2A3)
          IF (LV .LT. 0 .OR. JSIT(I) .EQ. '-99')GO TO 685
          NS=0
          NIP = 0
          DO 670 NZ=1,MSIT
            IF (JSIT(I).FQ.LABSI(NZ)) NS=NZ
            IF(SIP .EQ. LABSI(NZ)) NIP = NZ
  670     CONTINUE
          DO 680 KK=1,MXUNIT
            IF (IU(KK).EQ.JUN(I))  THEN
              ID=KK
              MUSIT(ID)  =NIP
              GO TO 690
            ENDIF
  680     CONTINUE
```

```
C
C  IF ALLOCATION READ FROM FILE 12 IS INFEASIBLE THIS YEAR,
C  RELABEL THE UNIT SET AND ALLOCATE IT TO THE CONUS SITE DURING
C  ASSESSMENT IN SUBROUTINE ALG SO THAT IT WILL BE IGNORED DURING
C  ASSESSMENT.
C
  685         JSIT(I) = '-01'
              JUN(I) = MXUNIT
              GO TO 700
  690         NUIS(NS)=NUIS(NS)+1
              NN=NUIS(NS)
              INPLU(NN,NS)=ID
              IF (NIP .EQ. 0)NIP = NS
              IUSIT(ID)=NIP
              JUN(I)=ID
  700       CONTINUE
C
C  BRANCH TO REWIND OF INPUT FILES
C
            GO TO 930
          ENDIF
C
C  IF THIS RUN IS IN ALLOCATION PHASE ,AND THIS IS THE SECOND
C  YEAR PROCESSED (MYR =1 WHICH IS THE 'ACTUAL YEAR 1', OR IF
C  THIS RUN IS THE BASELINE CASE (IN-PLACE ASSESSMENT ONLY)AND
C  THIS IS THE FIRST YEAR PROCESSED (NYR =1) THEN :
C
C  READ, FROM FILE 11, THE IDS OF UNIT SETS WHICH ARE IN PLACE AT
C  SPECIFIED SITES (AT START OF THE FIRST YEAR.).
C
C  STORE ALL THE UNIT SET IDS AT EACH IN-PLACE SITE IN ARRAY INPLU.
C  STORE THE IN-PLACE SITE FOR EACH SPECIFIED UNIT IN ARRAY IUSIT.
C  STORE THE TOTAL NUMBER OF UNITS IN-PLACE AT EACH SITE IN ARRAY NUIS.
C
C  ASSIGN ANY UNIT SET WITHOUT A SPECIFIED IN-PLACE SITE TO BE IN-PLACE
C  AT THE CONUS SITE, SITE -01 (WITH INDEX LSIT).
C
C  DURING ASSESSMENT PHASE (WHEN MYR = 1 AND NYR = 1) DO NOT REREAD IN-PLACE
C
          IF (IND5 .GE. 0 .AND. NYR .EQ. 1) GO TO 930
          NK = 0
C
C  DURING THE ALLOCATION PHASE, IN THE ACTUAL FIRST YEAR (YEAR 1)
C  STORE THE INDEX OF THE ACTUAL IN-PLACE SITE FOR UNIT W/INDEX ID
C  IN JUN(ID) SO THAT IT CAN BE READ IN THE ASSESSMENT PHASE
C  DURING THE BASELINE CASE, ALSO STORE IN-PLACE SITE INDICES IN JUN(ID)
C
          IF (NYR.EQ.1 .OR.  MYR .EQ. 1 )THEN
            DO 790 KS=1,NSIT
              READ (11,330,END=791) LAB
              NS=0
              DO 710 NZ=1,NSIT
                IF (LAB.EQ.LABSI(NZ)) NS=NZ
  710         CONTINUE
```

```
            IF (NS.EQ.0)  THEN
              WRITE (*,720) LAB
  720         FORMAT (1X,'FATAL ERROR: IN-PLACE SITE ',A5,
     +'  IS NOT DEFINED IN FILE 9')
              IF (IPR19 .NE. 0)WRITE (19,720) LAB
              IERR = 1
              GO TO 790
            ENDIF
            NN=0
  730       READ (11,90,END=780) LINPL
            IF (LINPL.LT.0) GO TO 780
            IF (LINPL.EQ.0) GO TO 730
C
C  NOTE ERROR IF MORE THAN MXLIST UNIT SETS ARE IN-PLACE AT A SITE.
C
            IF ((NN+1).GT.MXLIST)  THEN
              IERR = 1
              WRITE (*,740) LAB,MXLIST
  740         FORMAT (' FATAL ERROR: SITE',A4,' HAS MORE',
     +' THAN ',I4,' INPLACE UNITS')
              IF(IPR19 .NE. 0)WRITE (19,740) LAB,MXLIST
              IERR = 1
              GO TO 780
            ENDIF
            ID=0
            DO 750 KK=1,MXUNIT
              IF (IU(KK).EQ.LINPL)  THEN
                ID=KK
                IF(IND5 .EQ. 0)THEN
                  JUN(ID) = NS
                  GO TO 730
                ENDIF
                GO TO 770
              ENDIF
  750       CONTINUE
C
C  NOTE ERROR IF AN IN-PLACE UNIT SET WAS NOT DEFINED IN FILE 8.
C
C           WRITE (*,760) LINPL,LAB
C 760       FORMAT (' ERROR: UNIT',I5,'INPLACE AT SITE',A4,
C    + ' IS NOT DEFINED IN FILE 8')
C           IF(IPR19 .NE. 0)WRITE (19,760) LINPL,LAB
            GO TO 730
  770       IF (MUREQ(ID).LE.0.AND.MUWT(ID).LE.0) GO TO 730
            NN=NN+1
            INPLU(NN,NS)=ID
            IUSIT(ID)=NS
C
C  DURING BASELINE CASE (IN-PLACE ASSESSMENT),, SET SITE ASSIGNMENTS
C
                IF(IND5 .LT. 0)THEN
                  NK = NK+1
                  JUN(NK) = ID
                  JSIT(NK) = LAB
```

```
              ENDIF
            GO TO 730
  780       IF(IND5 .LT. 0)NUIS(NS)=NN
  790     CONTINUE
  791     IF (IND5 .EQ. 0)GO TO 801
C
C  ASSIGN UNITS NOT ASSIGNED AN IN-PLACE SITE TO THE CONUS SITE
C
          DO 800 I=1,MXUNIT
            IF ((MUREQ(I)+MUWT(I)).GT.0.AND.IUSIT(I).EQ.0)  THEN
              NUIS(LSIT)=NUIS(LSIT)+1
              NU=NUIS(LSIT)
              INPLU(NU,LSIT)=I
              IUSIT(I)=LSIT
                IF(IND5 .LT. 0)THEN
                  NK = NK+1
                  JUN(NK) = ID
                  JSIT(NK) = '-01'
                ENDIF
            ENDIF
  800     CONTINUE
          IF(IND5 .LT. 0)NCNT(1) = NK
        ENDIF
C
C  AT START OF EACH YEAR AFTER THE FIRST YEAR, THE ALLOCATIONS (UNIT SETS
C  /SITES) FROM THE PREVIOUS YEAR ARE MADE THE CURRENT IN-PLACE
C  UNIT SETS/SITES ASSIGNMENTS.
C
  801   IF (NYR .GT. 1)THEN
C
C  TRANSLATE UNIT SET INDEXES FROM THE PREVIOUS YEAR INTO UNIT SET
C  INDEXES FOR THE CURRENT YEAR.
C
          DO 820 L=1,LNTUN
            KTRANS(L)=0
            DO 810 K=1,NTUN
              IF (IIU(L).EQ.IU(K))  THEN
                KTRANS(L)=K
                GO TO 820
              ENDIF
  810       CONTINUE
C
C  ADD UNIT SETS PRESENT IN PREVIOUS YEAR BUT ABSENT THIS YEAR TO LIST
C  OF UNIT SETS IN CURRENT YEAR, BUT ASSIGN THEM A ZERO SIZE AND WEIGHT
C  IN THIS YEAR.
C
            NTUN=NTUN+1
            IU(NTUN)=IIU(L)
            IPUN(NTUN)=IIPUN(L)
            NPRI(NTUN)=INPRI(L)
            MUWT(NTUN)=0
            MUREQ(NTUN)=0
            KTRANS(L)=NTUN
  820     CONTINUE
```

```
         IF(NTUN .GT. MXUNIT)THEN
           IERR = 1
           WRITE(*,701)NTUN,MYR,MXUNIT
           IF(IPR19 .NE. 0)WRITE(19,701)NTUN,MYR,MXUNIT
  701      FORMAT(1X,' FATAL ERROR:',I6,' = NR OF UNITS THRU YR',I3,

     +     ' EXCEEDS',I6)
         ENDIF
C
C  TRANSLATE SITE INDEXES FOR THE PREVIOUS YEAR INTO SITE INDEXES FOR
C  THE CURRENT YEAR.
C
         DO 850 I=1,LNSIT-1
           DO 830 NS=1,NSIT-1
             ITRANS(I)=0
             IF (LABSI(NS).EQ.LLABSI(I))  THEN
               ITRANS(I)=NS
               GO TO 850
             ENDIF
  830      CONTINUE
           IF (ITRANS(I).EQ.0)  THEN
             WRITE (*,840) LABSI(I),NYR-1
  840        FORMAT (1X,'FATAL ERROR: ALLOC SITE ',A5,' IN YR',I3,
     +' DOES NOT EXIST',' NEXT YR')
             IF(IPR19 .NE. 0)WRITE (19,840) LABSI(I),NYR-1
           ENDIF
  850    CONTINUE
C
C  IF IHOLD(I) .GT. 0, THE RUN IS IN AN ALLOCATION PHASE,
C  AND UNIT SET I HAS IHOLD(I) AS ITS GOAL SITE, I.E. UNIT SET I
C  WAS ALLOCATED TO SITE IHOLD(I) WHERE IT WAS ALLOCATED IN THE
C  GOAL YEAR (WHEN RUN AS THE INITIAL YEAR PROCESSED).
C  (IF UNIT I WAS NOT PRESENT IN THE LAST YEAR, ITS GOAL
C  SITE WAS DETERMINED IN THE FIRST YEAR IT APPEARED.)
C  TRANSLATE PREVIOUS YEAR'S INDEXES OF GOAL SITES INTO INDEXES
C  FOR THE CURRENT YEAR.
C
         DO 870 IZ=1,MXHOLD
           IF (IHOLD(IZ).GT.0)  THEN
             NZ=IHOLD(IZ)
             IHOLD(IZ)=ITRANS(NZ)
             NM=IHOLD(IZ)
           ENDIF
  870    CONTINUE
C
C  TRANSLATE INDEXES OF PREVIOUS YEAR'S ALLOCATION SITES INTO INDEXES FOR
C  THE CURRENT YEAR.
C
         DO 890 NS=1,LNSIT-1
           NSZ=ITRANS(NS)
           NU=0
           MM=MINP(NS)
           IF (MM.GT.0)  THEN
             DO 880 JK=1,MM
```

```
                IF (NALLU(NS,JK).NE.0)  THEN
                  NU=NU+1
                  IZ=NALLU(NS,JK)
                  INPLU(NU,NSZ)=ktrans(IZ)
                  ID=INPLU(NU,NSZ)
                  IUSIT(ID)=NSZ
                ENDIF
 880          CONTINUE
            ENDIF
            NUIS(NSZ)=NU
 890      CONTINUE
          IF (MINP(LMSIT).GT.0)  THEN
            NU=0
            DO 900 JK=1,MINP(LMSIT)
              IF (NALLU(LMSIT,JK).NE.0)  THEN
                NU=NU+1
                IZ=NALLU(LMSIT,JK)
                INPLU(NU,MSIT)=KTRANS(IZ)
                ID=INPLU(NU,MSIT)
                IUSIT(ID)=MSIT
              ENDIF
 900        CONTINUE
          ENDIF
          NUIS(MSIT)=NU
          DO 920 I=1,mxunit
            IZ=IU(I)
            IF ((MUREQ(I)+MUWT(I)).GT.0.AND.IUSIT(I).LE.0) THEN
C
C  IN THE ALLOCATION PHASE, IF A UNIT I WAS NEVER ASSIGNED A
C  GOAL SITE (IHOLD(I)) IN THE INITIAL YEAR, AND
C  IS APPEARING THIS YEAR FOR THE FIRST TIME, THEN PRINT A
C  WARNING THAT THE UNIT WAS NOT PRESENT IN THE FINAL YEAR.
C
              NUIS(LSIT)=NUIS(LSIT)+1
              NU=NUIS(LSIT)
              INPLU(NU,LSIT)=I
              IUSIT(I)=LSIT
         IF (IND5 .EQ. 0 .AND. IHOLD(IZ) .EQ. 0.AND. IPR19 .NE. 0)THEN
                WRITE (*,910) IZ,MYR
 910            FORMAT (1X,'*** WARNING - UNIT',I4,' IN YR',I3,
     +          ' IS NOT PRESENT IN THE FINAL YEAR ***')
                IF(IPR19 .NE. 0)WRITE (19,910) IZ,MYR
              ENDIF
            ENDIF
 920      CONTINUE
          DO 921 NN = 1,NSIT
           IF (NUIS(NN) .GT. MXLIST)THEN
             IERR = 1
             WRITE(*,922)LABSI(NN),NUIS(NN),MXLIST
             IF(IPR19 .NE. 0) WRITE(19,922)LABSI(NN),NUIS(NN),MXLIST
 922         FORMAT (' FATAL ERROR: SITE',A4,' HAS',I4,
     +' UNITS IN-PLACE - EXCEEDING',I4)
           ENDIF
 921       CONTINUE
```

```
C
C  IN THE ALLOCATION PHASE, AFTER FIRST(GOAL) YEAR ONLY (IN WHICH DATA FOR
C  THE GOAL YEAR WAS USED) REWIND ALL INPUT FILES TO BE ABLE
C  TO READ THE YEAR 1 INPUT NEXT.
C
      ENDIF
  930 IF (ISTOP.EQ.0)  THEN
        rewind 8
        rewind 9
        rewind 10
        rewind 11
        MYR=9
        istop=1
      ENDIF
C
C  ABORT RUN IF A FATAL ERROR WAS DETECTED
C
      IF (IERR.NE.0) GO TO 1250
      DO 940 I=1,mxunit
        IF (IU(I).GT.0)  THEN
          IZ=IU(I)
          LMUREQ(IZ)=MUREQ(I)+MUWT(I)
          IF(IND5 .LT. 0 .AND. NYR .EQ. 1)LLMURQ(I) = LMUREQ(IZ)
        ENDIF
  940 CONTINUE
      DO 960 NS=1,MSIT
        NSPOC(NS)=0
        DO 950 K=1,MXPKG
  950   ITOT(K)=0
  960 CONTINUE
C
C  IF THIS IS AN ASSESSMENT PHASE OR IS THE BASELINE CASE,
C  BRANCH TO CALL FOR SUBROUTINE ALG
C
      IF (IND5 .NE. 0)GO TO 1205
C
C  MARK ALL UNITS WITH SIZE = WEIGHT = 0 AS UNAVAILABLE FOR ALLOCATION.
C
      DO 970 K=1,mxunit
        IF (IU(K).GT.0)  THEN
          IZ=IU(K)
          MOF(K)=0
          IF (LMUREQ(IZ).GT.0) MOF(K)=1
        ENDIF
  970 CONTINUE
C
C  ORDER SITES BY DECREASING VALUE OF SITE CAPACITY AND DETERMINE
C  THE REDUNDANT SITES AT THE END OF THIS ORDERED LIST.  RETAIN
C  UNALTERED THE FIRST NSLIM ORDERED SITES ON THIS LIST AND SET ALL
C  SUCCEEDING (ON ORDERED LIST) SITE CAPACITIES TO ZERO.
C
      NKNT=0
      DO 980 NS=1,NSIT-1
        IF (NSCAP(NS).GT.0)  THEN
```

```
              NKNT=NKNT+1
              IX(NKNT)=ISPR(NS)
              INDS(NKNT)=NS
            ENDIF
  980     CONTINUE
          CALL ORDER (NKNT)
          IF (IBUGYR.NE.0 .AND. IND5 .EQ. 0 )  THEN
            IF(IPR19 .NE. 0)WRITE (19,990) MYR
  990       FORMAT (//,2X,'ORDERED CUMULATIVE NONZERO SITE CAPACITIES  ***
     +',' YEAR =',I2)
            IF(IPR19 .NE. 0)WRITE (19,1000)
 1000       FORMAT (//,1X,' SITE     CAPACITY CUMULATED    NEEDED',/)
          ENDIF
          IOUT=0
          ISI=0
          NTC=0
          ICH=0
          DO 1020 NS=1,NKNT
            LL=INDS(NS)
            MARKS=1
            IF (NSCAP(LL).LE.0) MARKS=0
            NTC=NTC+NSCAP(LL)
            IF (IPAS .EQ. 2) NTC = 99999999
            IF (NTC.GT.MALL.AND.ICH.EQ.1) MARKS=0
            IF (NTC.GT.MALL) ICH=1
            IF (MARKS.EQ.0.AND.NSCAP(LL).GT.0) IOUT=IOUT+1
            IF (IOUT.GT.NSLIM) NSCAP(LL)=0
            IF (IBUGYR.NE.0 )
     +      WRITE(19,1010) LABSI(LL),NSCAP(LL),NTC,MARKS
 1010       FORMAT (2X,A5,2X,3I10)
            ISI=ISI+MARKS
 1020     CONTINUE
          IF (IBUGYR.NE.0 )  THEN
            WRITE (19,1030) ISI
 1030       FORMAT (/,1X,' EXACTLY',I4,' SITES ARE NEEDED IN THIS PROBLEM'
     +,/)
            WRITE (19,1040) IOUT
 1040       FORMAT (/,1X,' EXACTLY',I4,' SITES W/NONZERO CAPACITY ARE',' R
     +EDUNDANT',/)
            NDP = NTC - MALL
            WRITE(19,1046)MYR,MALL,NTC
 1046       FORMAT(/,1X,'IN YR',I3,' TOT UNIT AREA =',I8,
     +      ' TOT STORAGE CAPACITY =',I8,/)
            IF (NDP .LT. 0)THEN
              WRITE(*,1041)MYR,-NDP
              WRITE(19,1041)MYR,-NDP
 1041         FORMAT(/,1X,'WARNING: INSUFFICIENT TOTAL SITE CAPACITY IN',
     +        ' YR',I3,/,' AT LEAST',I7,' MORE STORAGE IS NEEDED',/)
            ENDIF
          ENDIF
          IF (IPR16.EQ.0.AND.MYR.EQ.1)  THEN
            DO 1070 NS=1,NSIT-1
              DO 1060 NC=1,NSIT-1
                ZZ = FLOAT(JDIST(NS,NC)/10.)
```

```
            WRITE (16,1050) LABSI(NS),LABSI(NC),ZZ
1050        FORMAT (2A3,F7.1)
1060      CONTINUE
1070    CONTINUE
      ENDIF
      IF (IPR16 .NE. 0)  THEN
        WRITE(16,1172) MYR
1172 FORMAT(/,1X,'DISTANCES (KM) BETWEEN SITES IN YEAR',I3,/
    +,1X,'SITE SITE  DIST(KM)')
        DO 1071 NS=1,NSIT-1
          DO 1061 NC=1,NSIT-1
             ZZ = FLOAT(JDIST(NS,NC)/10.)
            WRITE (16,1051) LABSI(NS),LABSI(NC),ZZ
1051        FORMAT (2X,A3,2X,A3,F9.1)
1061      CONTINUE
1071    CONTINUE
      ENDIF
C
C  IN THE ALLOCATION PHASE, FOR EACH PROJECT IN THIS YEAR,
C  CONSTRUCT AN ORDERED PREFERENCE LIST OF SITES. THIS LIST IS ORDERED
C  BY DECREASING PREFERENCE FOR ALLOCATING THE PROJECT TO THAT SITE.
C  THE PREFERENCE MOE FOR A PROJECT/SITE COMBINATION
C  IS BASED ONLY ON THE AVERAGE CLOSENESS  OF UNIT SET GDPS (IN THE
C  PROJECT) TO THAT SITE.  THE AVERAGE IS OVER ALL UNIT SETS IN THE
C  PROJECT.
C
       IF (IHD .NE. 0 .AND. IPR18 .NE. 0)
    +   WRITE(18,1105)MYR
1105    FORMAT (/,'ORDERED  PREFERENCE LIST OF SITES FOR PROJECTS',
    +' IN YEAR',I3,//)
      DO 1130 N=1,MXPKG
        IF (IPRIO(N).EQ.0) GO TO 1130
        NX=NUNP(N)
        IF (NX.LE.0) GO TO 1130
        NST=0
        DO 1090 K=1,NSIT-1
          IF (NSCAP(K).LE.0) GO TO 1090
          NST=NST+1
          ISUM=0
          JWT=1.
          DO 1080 I=1,NX
            ID=IDPKG(N,I)
            IF (ID.GT.0 )
    +       ISUM=ISUM+IDIST(K,ID)*MUWT(ID)
            JWT=JWT+MUWT(ID)
1080      CONTINUE
          IX(NST)=ISUM/JWT
          INDS(NST)=K
1090    CONTINUE
        NTOX(N)=NST
        IF (NST.GT.0) CALL ORDER (NST)
       IF (IHD .NE. 0 .AND. IPR18 .NE. 0)
    +   WRITE(18,1100)
1100    FORMAT (/,1X,'SITE    KM/TON TO GDP',6X,'PROJ   YR')
```

```
          DO 1120 K=1,NST
            ZZ = FLOAT(IX(K)+5)/10.
            IF(IPR18 .NE. 0)WRITE (18,1110) LABSI(INDS(K)),ZZ,
     +      N,MYR
 1110       FORMAT (1X,A4,7X,F10.1,I10,I5)
 1120     ISTO(N,K)=INDS(K)
          IF (IND5 .EQ. 0 .AND. IBUGYR .NE. 0)
     +    WRITE(19,611)MYR,N,MPREQ(N),JWT
  611     FORMAT(1X,'YR=',I3,' PROJ=',I3,' SIZE =',I8,' WT=',F14.0)
 1130   CONTINUE
C
C  IN THE ALLOCATION PHASE, FOR EACH UNIT SET IN THIS YEAR, CONSTRUCT
C  AN ORDERED PREFERENCE LIST OF SITES. THIS LIST IS ORDERED BY DECREASING
C  PREFERENCE FOR ALLOCATING THE UNIT SET TO THAT SITE.
C  THE PREFERENCE FOR A SITE BY A UNIT SET IS BASED ON
C  THE TON-MILE SEPARATION BETWEEN THE SITE AND THE UNIT SET GDP.
C
          IF (IHD .NE. 0 .AND.ICYR .NE. 0) WRITE(21,1153)MYR
 1153     FORMAT(//,1x,' SITE PREFERENCE LIST FOR UNITS IN YR',I3,//)
        DO 1170 I=1,MXUNIT
          IZ=IU(I)
          IF (IZ.LE.0) GO TO 1170
          IF (MUREQ(I).LE.0.AND.MUWT(I).LE.0) GO TO 1170
          IF (IZ.LE.0) GO TO 1170
          LP=IPUN(I)
          IP=IUSIT(I)
          IF (IP .GT. NSIT)IP = NSIT
          NST=0
          DO 1140 NS=1,NSIT-1
            IF (NSCAP(NS).LE.0.AND.NYR.NE.1) GO TO 1140
            NST=NST+1
           IF (IP.LE.0.OR.IP.GT.MXSITE)  THEN
              IWW(NS)=0
            ELSE
              ZZ = FLOAT(MUWT(I))*FLOAT(JDIST(IP,NS))
              IWW(NS)=ZZ
            ENDIF
            IX(NST) = IDIST(NS,I)
            INDS(NST) = NS
 1140     CONTINUE
          CALL ORDER (NST)
C
C  IN THE ALLOCATION PHASE, INITIALIZE EACH UNIT SET'S GOAL SITE TO THE
C  CLOSEST OPEN SITE TO THAT UNIT SET'S GDP LOCATION.
C
            IF (IHOLD(IZ).EQ.0.AND.IMIN(I).GT.0) IHOLD(IZ)=IMIN(I)
          SUMJ=0.
          AVG=0
          IF (MYR .EQ. ICYR .OR. ICYR .EQ. 99)THEN
            IF (IHD .NE. 0 .AND.ICYR .NE. 0) THEN
            WRITE(21,1155)
 1155     FORMAT(/,16X,'KM TO ',/,1X,'UNIT  SITE      GDP  YR')
            ENDIF
            DO 1160 K=1,NST
```

```
              ZZ = FLOAT(IDIST(INDS(K),I))/10.
              IF (MUWT(I) .GT. 0 .AND. IHD .EQ. 0)
     +        WRITE (21,1150) IU(I),LABSI(INDS(K)),
     +        IDIST(INDS(K),I),MYR
 1150         FORMAT (1X,I5,A5,F10.1,I4)
              IF (MUWT(I) .GT. 0 .AND. IHD .NE. 0)
     +        WRITE (21,1150)IU(I),LABSI(INDS(K)),
     +        ZZ,MYR
 1160       CONTINUE
          ENDIF
 1170   CONTINUE
        DO 1190 NS=1,MSIT
          DO 1180 JK=1,MXLIST
 1180     NALLU(NS,JK)=0
 1190   MINP(NS)=0
C
C  IN THE ALLOCATION PHASE, CALL ROUTINE BYUNIT TO MARK
C  UNIT SETS ARE IN PLACE AT THEIR GOAL SITES AND WHICH CAN AND
C  WILL BE ALLOCATED THERE (AT THEIR IN-PLACE GOAL SITES) BEFORE
C  OTHER UNIT SET ALLOCATIONS ARE DONE.
C
        IF (IND5.EQ.0)  THEN
          IF (NYR.GT.1.AND.NYR.LT.NFIN)  THEN
            DO 1200 NS=1,NSIT-1
              IF (NUIS(NS).GT.0) CALL BYUNIT (NS)
 1200       CONTINUE
          ENDIF
        ENDIF
C
C  CALL ROUTINE ALG AND THE MAIN ALLOCATION ALGORITHMDS.
C
 1205   CALL ALG (MSPOC)
C
C  DO AN OVERALL ALLOCATION SUMMARY (FILE 17) DURING THE ASSESSMENT
C  PHASE BUT DO NOT ASSESS THE GOAL YEAR WHEN IT IS THE FIRST YEAR
C  PROCESSED (NYR = 1). (IT IS ASSESSED ONLY WHEN IT IS THE LAST
C  YEAR PROCESSED (NYR = NFIN + 1)).
C
        IF (IND5.EQ.0.AND.NYR.EQ.1) GO TO 1240
        IST=0
        DO 1210 NS=1,MSIT
          IF (NSPOC(NS).GT.0)  THEN
            IST=IST+1
          ENDIF
 1210   CONTINUE
        IF (NB .EQ. 0)NB = -1
        IF (IMTOT.EQ.0) IMTOT=-999.
        IF (MPCT.EQ.0) MPCT=1000
        IF (IND5 .LT. 0)ZPKG = ZTP
        ZMPC=MPCT+(ZPKG-NB)
        FX=ZMPC/ZPKG
        Z5 = IST
        Z4 = TMOV/FLOAT(MNUN)
        IF (NB .EQ. -1) FX = 1.
```

```
       IF (IND5 .NE. 0 .AND. IPR17 .NE. 0)
     + WRITE (17,1220) MYR,XDG/(10.*MNUN),FX,Z5,Z4,MNUN
 1220  FORMAT (1X,'AV=',I3,4X,10X,F10.0,F10.2,F10.0,F5.2,I9)
      IF (IND5.EQ.0 .AND. MYR .NE. 9)THEN
       WRITE (*,1224)MYR
 1224  FORMAT (/,1X,' ** SITER ALLOCATIONS COMPLETED IN YEAR',I3)
       IF (IBUGYR .NE. 0)WRITE(19,1225)IMALL,MALL
 1225  FORMAT(/,1x,' TOTAL ALLOCATED SPACE =',I12,' OUT OF',I12)
      ENDIF
      IF (JFUL .NE. 0)THEN
       WRITE(*,1226)MYR,MXLIST
       IF(IPR19 .NE. 0)WRITE(19,1226)MYR,MXLIST
 1226  FORMAT(1X,'WARNING : YEAR'I3,' ALLOCATIONS ARE INVALID',
     +/,' BECAUSE NUMBER OF UNITS ALLOCATED TO A SITE EXCEED',I4)
       ENDIF
       DO 1238 k=1,MXUNIT
 1238  LLMURQ(K) = MUREQ(K) + MUWT(K)
 1240 CONTINUE
 1250 CONTINUE
      IF (IND5.EQ.0 )THEN
       WRITE (*,1260)
       IF(IPR19 .NE. 0)WRITE(19,1260)
 1260  FORMAT (/,1X,' **  ALLOCATIONS COMPLETED FOR ALL YEARS ** ')
      ELSE
       WRITE (*,1270)
       IF(IPR19 .NE. 0)WRITE(19,1270)
 1270  FORMAT (/,1X,' ** SITER ASSESSMENT COMPLETED **')
      ENDIF
C
C  IF ASSESSMENT PHASE IS DONE, THEN QUIT.
C  IF ALLOCATION PHASE HAS JUST FINISHED, RESET NFIN TO ACTUAL NUMBER
C  OF YEARS IN TIMEFRAME AND BRANCH TO START OF MAIN PROGRAM
C
      IF (ISK .EQ. 1 .OR. IPR17 .EQ. 0)GO TO 1199
      IF (IPAS .LE. 2)THEN
       REWIND  8
       REWIND  9
       REWIND 10
       REWIND 11
       REWIND 12
       REWIND 15
       REWIND 16
       ISK = 1
       NFIN = NFIN - 1
       GO TO 776
      ENDIF
 1199    END
```

```
$LARGE
      SUBROUTINE ALG (MSPOC)
C
C  THIS ROUTINE CALLS THE MAIN ALLOCATION ROUTINE DURING THE ALLOCATION
C  PHASE.  DURING THE ASSESSMENT PHASE, IT SETS VARIABLES AND CALLS
C  THE MAIN ASSESSMENT ROUTINE.
C
$INCLUDE:'COMMON.INC'
$INCLUDE:'NEWCOM.INC'
      NRP=MXPKG
      IEXIT=0
C
C  LOOP FOR PROCESSING PROJECTS DURING THE ASSESSMENT PHASE.
C  (IN ALLOCATION PHASE THE LOOP IS EXITED AFTER ONE CYCLE.)
C
      IF(IND5 .LT. 0)MNUN = 0
      DO 180 IB=1,MXPKG
C
C  IN ALLOCATION PHASE, CALL THE INPUT DISPLAY ROUTINE (DISPIN) AND
C  THE MAIN ALLOCATION ROUTINE (BYUNIT).
C
        IF (IND5.EQ.0)  THEN
          IG=0
          DO 20 NS=1,NSIT
            NU=NUIS(NS)
   20     CONTINUE
          IREG=0
          CALL BYUNIT (0)
          IEXIT=1
          GO TO 50
        ENDIF
        IF (IPR(IB).LE.0) GO TO 180
        IG=IPR(IB)
C
C  IN ASSESSMENT PHASE SET ALLOCATION STATUS VARIABLES PRIOR TO ASSESSMENT
C
        IF (IND5.NE.0)  THEN
          IF (IPRIO(IG) .LT. 0)GO TO 180
          DO 40 NS=1,NSIT -1
            MINP(NS)=0
            DO 30 K=1,NCNT(MYR)
              ID=JUN(K)
              IF (JSIT(K).EQ.LABSI(NS).AND.IPUN(ID).EQ.IG)  THEN
C
C  IN ASSESSMENT PHASE, SET MINP(NS) = NR OF UNIT SETS IN PROJECT IG WHICH
C  ARE ALLOCATED TO SITE NS.
C
                MINP(NS)=MINP(NS)+1
                NSPOC(NS)=NSPOC(NS)+MUREQ(ID)
                LL=IPUN(ID)
                ITOT(LL)=ITOT(LL)+MUWT(ID)
                JS=IUSIT(ID)
```

```
              IWTMV(ID)=MUWT(ID)*JDIST(JS,NS)
              NDIS(ID)=(IDIST(NS,ID)-LDIS(ID))*MUWT(ID)
              NALLU(NS,MINP(NS))=ID
            ENDIF
   30     CONTINUE
   40   CONTINUE
      ENDIF
   50 IREG=1
C
C  IN THE ALLOCATION PHASE, ALLOCATE UNITS NOT PRESENT THIS YEAR (THOSE
C  NONZERO SIZE OR WEIGHT) TO THEIR GOAL SITES.
C
      IF (IND5.EQ.0)  THEN
        DO 60 K=1,MXUNIT
          IZ=IU(K)
          IF (IZ.LE.0) GO TO 60
          IF (IZ.GT.0.AND.IHOLD(IZ).NE.0.AND.LMUREQ(IZ).EQ.0)  THEN
            NS=IHOLD(IZ)
            MINP(NS)=MINP(NS)+1
            JK=MINP(NS)
            NALLU(NS,JK)=K
            MOF(K)=2
          ENDIF
   60   CONTINUE
      ENDIF
C
C  IN REST OF ROUTINE, WRITE ALLOCATION SUMMARIES AS INDICATED BY INPUT.
C  CALL THE MAIN ALLOCATION ASSESSMENT ROUTINE.
C
      CALL DISPAL (IG,MPC,MSPOC,IMAXP,XKSUM,ITW,ITS)
      IF (MPC.GE.2.AND.IG.GT.0)  THEN
        NB=NB+1
      MPCT=MPCT+MPC
      ENDIF
      XDG=XDG+XKSUM*NUMP
      IF (IND5.NE.0) NTOT=ITOT(IG)
      IF (IND5.EQ.0) NTOT=LTOT
      IMALL=IMALL+ITS
      IMTOT=IMTOT+ITW
      MPR=MALL
      IF (IND5.NE.0) MPR=MPREQ(IG)
      IF (IBUGYR.NE.0) WRITE (19,130) ITS,MPR
  130 FORMAT (/,1X,' TOTAL ALLOCATED SPACE =',I12,5X,'OUT OF',I12,/)
      IF (IBUGYR.NE.0) WRITE (19,140) ITW,NTOT
  140 FORMAT (/,1X,' TOTAL ALLOCATED NON-OVERFLOW WEIGHT =',I12,5X,'OU
     +T OF',I12)
      IYY=XKSUM*NUMP
      IF (IBUGYR.NE.0) WRITE (19,160) IYY
  160 FORMAT (/,1X,' TOTAL WT-KM MOVED : ALLOC SITE TO GDP =',I12)
      MF = 0
      IF(MPC .GT. 0)MF = MPREQ(IG)/MPC
      IF (MYR.NE.9.AND.IND5.NE.0)  THEN
        Z1=IMAXP
        Z5 = MPC
```

```
          Z2 = MPREQ(IG)
          IF (Z2 .LT. .001)Z2=-1.
          INU = NUNP(IG)
          IF (IND5 .LT. 0)THEN
            INU = 0
            DO 162 L = 1,NUNP(IG)
              IL = IDPKG(IG,L)
              IF (IUSIT(IL) .GT. 0.AND.IUSIT(IL) .LT. (MSIT-1))
     +        INU = INU+1
162         CONTINUE
            MNUN = MNUN + INU
            IF (INU .GT. 0) ZTP = ZTP + 1.
        ENDIF
          IF(IPR17 .NE. 0)
     +    WRITE (17,170) MYR,IG,Z1/Z2,XKSUM/10.,Z5,Z5,
     +    ZNUC(MYR,IG),INU
170       FORMAT (4X,I3,I4,F10.2,F10.0,F10.2,F10.0,F5.2,I9)
        ENDIF
        IF (MYR .NE. 9 .AND. IND5 .EQ. 0)THEN
          DO 161 K = 1,MXPKG
            IF(IPRIO(K) .EQ. 0)GO TO 161
          Z2 = NUNP(K)
          IF (Z2 .LT. .001)Z2=-1.
            ZNUC(MYR,K) = FLOAT(NUCP(K))/Z2
161       CONTINUE
        ENDIF
        IF (IEXIT.NE.0) RETURN
180   CONTINUE
      RETURN
      END
```

```
$LARGE
      SUBROUTINE BYUNIT (ISIT)
C
C  DURING ALLOCATION PHASE ONLY, THIS ROUTINE IS CALLED IN TWO DIFFERENT
C  WAYS.  FIRST, IN EACH YR AFTER THE FIRST YEAR PROCESSED(THE GOAL YEAR)
C  AND BEFORE THE LAST YEAR PROCESSED (THE GOAL YEAR AGAIN), IT IS CALLED IN
C  A 'MARK UNITS'(ISIT .GT. 0) CYCLE FOR EACH SITE ISIT. LATER IN THAT YEAR,
IT
C  IS CALLED ONCE IN AN 'ALLOCATION CYCLE'(ISIT = 0) IN WHICH MOST UNITS
C  ARE ALLOCATED.  IN THE FIRST YR PROCESSED AND THE LAST YEAR PROCESSED
C  (WHICH ARE BOTH THE GOAL YEAR -  ACTUALLY THE LAST YEAR OF THE TIMEFRAME)
C  ONLY THE 'ALLOCATION CYCLE IS CALLED AND IT IS CALLED EXACTLY ONCE.
C  IF THIS IS A 'MARK UNITS' CALL (ISIT .GT. 0), THE ROUTINE FIRST ALLOCATES
:
C   1. ALL DESIGNATED PROJECT ALLOCATIONS, IF ANY
C   2. ALL PROJECTS WHICH CAN BE ALLOCATED INTACT, IF THIS IS 'REDUCE
DISPERSION'
C       OPTION
C   3. ALL DESIGNATED UNIT ALLOCATIONS, IF ANY
C
C  AFTER THE ABOVE ALLOCATIONS ARE DONE, THE ROUTINE MARKS UNIT SETS IN-PLACE
C  AT A SPECIFIC SITE (ISIT) FOR RETENTION (FIXED ALLOCATION) THIS YEAR AT
C  THEIR GOAL SITES WHICH ALSO HAPPEN TO BE THEIR PREVIOUS YEAR'S ALLOCATION
C  SITES, BUT IT DOES NOT ALLOCATE THESE UNTIL THE ALLOCATION CYCLE WHICH
OCCURS
C  LATER IN THE SIMULATION.
C  IF THIS IS AN 'ALLOCATION CYCLE' CALL (ISIT = 0), THEN THIS ROUTINE
C  ORDERS UNIT SETS FOR ALLOCATION AND THEN ALLOCATES THE UNALLOCATED UNIT
SETS
C  TO SITES.
C
$INCLUDE:'COMMON.INC'
      DIMENSION
     +    IXO(MXPKG), IUN(MXUNIT),   MINDS(MXUNIT), MIX(MXUNIT)
C
C  ALL UNIT SETS (NTUN OF THEM) WILL BE ORDERED IF ALLOCATION IS TO BE
C  DONE ( ISIT .LE. 0).  ONLY THE UNIT SETS IN-PLACE AT SITE ISIT
C  [NUIS(ISIT) OF THEM] WILL BE ORDERED IF ONLY MARKING OF UNIT SETS
C  AT A SITE IS TO BE DONE IN THE ALLOCATION PHASE (ISIT .GT. 0).
C
      IL2=NTUN
      IF (ISIT.GT.0) IL2=NUIS(ISIT)
      IF (IL2.LE.0) RETURN
C
C  CHANGE THE INTERNAL UNIT SET PRIORITY OF THOSE UNIT SETS WHICH HAVE
C  BEEN MARKED (ISET(ID) .GT. 0) TO BE RETAINED AT GOAL SITES WHICH ARE
C  ALSO THEIR PREVIOUS YEAR'S ALLOCATION SITES.  CHANGE THESE UNIT SET
C  PRIORITIES SO THAT ALL MARKED UNIT SETS HAVE HIGHER PRIORITY
C  THAN ALL UNMARKED UNIT SETS, AND SO THAT THE PRIORITY ORDER OF
C  MARKED UNIT SETS WITH THE NEW PRIORITIES HAS THE SAME SEQUENCE AS
C  WOULD RESULT USING THE UNCHANGED INTERNAL UNIT SET PRIORITIES.
C
      NST = 0
```

```
      DO 10 K=1,IL2
        ID=K
        IF (ISIT.GT.0) ID=INPLU(K,ISIT)
C
C  THE UNMARKED UNIT SETS WILL BE ORDERED ACCORDING TO INTERNAL UNIT
C  PRIORITY (WHICH MERGED INPUT UNIT PRIORITY WITH INPUT PROJECT PRIORITY).
C  IN ALL CASES, THE MARKED UNIT SETS (ISET(ID) .NE. 0) WILL BE ORDERED
C  BY UNIT SET PRIORITY MODIFIED AS NOTED ABOVE.
C
        NST = NST + 1
        IX(NST) = NPRI(ID)
        IF (ISET(ID).NE.0) IX(NST)=-999999999+NPRI(ID)
   10 INDS(NST)=ID
      IL2 = NST
C
C  ORDER THE UNIT SETS ACCORDING TO THE SPECIFIED CRITERIA.
C
      CALL ORDER (IL2)
      DO 20 K=1,IL2
        MIX(K)=IX(K)
   20 MINDS(K)=INDS(K)
      IF (IL2.EQ.1) GO TO 70
      KNT=1
      LAST=MIX(1)
      DO 60 K=2,IL2
        IF (K.EQ.IL2) GO TO 30
        IF (MOF(MINDS(K)).EQ.0) GO TO 60
        IF ((MUREQ(MINDS(K))+MUWT(MINDS(K))).LE.0) GO TO 60
        LK=K
        IF (MIX(K).EQ.LAST)  THEN
          KNT=KNT+1
          GO TO 60
        ENDIF
C
C  BREAK TIES BY ORDERING TIED UNIT SETS ACCORDING TO INCREASING UNIT
C  SET SIZE.
C
   30   IF ((MIX(K).NE.LAST.OR.K.EQ.IL2).AND.(KNT.GT.1))  THEN
          MK=K
          IF (K.EQ.IL2) MK=LK+1
          DO 40 L=1,KNT
            IX(L)=MUREQ(MINDS(MK-L))
   40     INDS(L)=MINDS(MK-L)
          CALL ORDER (KNT)
          DO 50 L=1,KNT
   50     MINDS(MK-L)=INDS(KNT-L+1)
        ENDIF
        KNT=1
        LAST=MIX(K)
   60 CONTINUE
C  FOLLOWING CODE IS USEFUL FOR DEBUGGING
C     IF (IBUG6.NE.0)  THEN
C       WRITE (19,80)
C  80   FORMAT (//,1X,'1. UNIT SET MOES/PRIORITIES  2. UNIT SET ORDER :'
```

```
C    +)
C       WRITE (19,90)
C  90   FORMAT (1X,'--- LIST FOLLOWS ---')
C       WRITE (19,100) (MIX(K),K=1,IL2)
C 100  FORMAT (1X,'ORDERED IX  ',5I9)
C       WRITE (19,90)
C       WRITE (19,110) (MINDS(K),K=1,IL2)
C 110   FORMAT (1X,'ORDERED IND ',5I9)
C     ENDIF
C
C  ORDER PROJECTS ACCORDING TO PROJECT PRIORITY.  THEN, IN PRIORITY ORDER,
C  CHECK EACH PROJECT AGAINST THE DESIGNATED PROJECT ALLOCATION INPUT
C  (FILE 7) TO SEE IF THAT LIST SPECIFIES THIS PROJECT TO BE FIXED
C  AS AN UNBROKEN PROJECT ALLOCATION AT A SPECIFIED SITE THIS YEAR.
C  IF SO, ATTEMPT TO ALLOCATE THIS PROJECT AS DIECTED BY FILE 7
C  BEFORE THE NORMAL ALLOCATIONS OCCUR.
C
  70  NST=0
     DO 180 I=1,MXPKG
       IXO(I) = 0
       IF (IPRIO(I).NE.-998)  THEN
         NST=NST+1
         IX(I)=IPRIO(I)
         INDS(I)=I
       ENDIF
 180 CONTINUE
     CALL ORDER (NST)
     IF (MFL.GT.0.AND.IDUNP.EQ.0)  THEN
       IF (NST.GT.0.AND.MFL.GT.0)  THEN
         IDUNP=1
         DO 240 I=1,NST
           IF (MPREQ(INDS(I)).LE.0) GO TO 240
           DO 230 J=1,MFL
             IF (IFPN(J).LE.0) GO TO 230
             IF (IFPN(J).EQ.INDS(I).AND.(IFPY(J).EQ.MYR.OR.IFPY(J).EQ.9
    +        9.OR.(MYR.EQ.9.AND.IFPY(J).EQ.(NFIN-1))))  THEN
               NN=IFPN(J)
               DO 220 L=1,NSIT-1
                 IF (IFPS(J).EQ.LABSI(L))  THEN
                   INS=L
                   IF(NSCAP(INS) .LE. 0) GO TO 225
                   IF (MPREQ(NN).LE.(NSCAP(INS)-NSPOC(INS)))  THEN
                     IXO(NN) = 1
                     DO 190 KK=1,NUNP(NN)
                       ID=IDPKG(NN,KK)
 190                 CALL IALLUN (ID,INS)
                     WRITE (*,200) IFPN(J),MYR,IFPS(J)
 200                 FORMAT (1X,'DESIGNATION OF PROJECT',I5,' IN YR',I3
    +                ,' TO',' SITE ',A3)
                    IF(IPR19 .NE. 0)WRITE (19,200) IFPN(J),MYR,IFPS(J)
                     GO TO 240
                   ENDIF
 225               WRITE (*,210) IFPN(J),MYR,IFPS(J)
 210               FORMAT (1X,'UNABLE TO DESIGNATE PROJECT',I5,' IN YR'
```

```
     +,I3,' TO SITE ',A3)
                    IF(IPR19 .NE. 0)WRITE (19,210) IFPN(J),MYR,IFPS(J)
                    GO TO 230
                  ENDIF
 220            CONTINUE
                WRITE (*,210) IFPN(J),MYR,IFPS(J)
                IF(IPR19 .NE. 0)WRITE (19,210) IFPN(J),MYR,IFPS(J)
                GO TO 240
              ENDIF
 230        CONTINUE
 240      CONTINUE
        ENDIF
      ENDIF
C
C  CHECK THE DESIGNATED UNIT SET ALLOCATION INPUT (FILE 7).  ATTEMPT TO
C  PROCESS EACH DESIGNATED UNIT SET ALLOCATION FOR THIS YEAR BY
C  ATTEMPTING TO ALLOCATE THE DESIGNATED UNIT SET TO THE DESIGNATED SITE.
C
      IF (MFL.EQ.0.OR.IDUN.GT.0) GO TO 340
      IDUN=1
      DO 330 J=1,MFL
        IF (IFUN(J).LE.0) GO TO 330
        DO 320 IPIK=1,NTUN
          IF (MOF(IPIK) .EQ. 0)GO TO 320
          IF ((MUREQ(IPIK)+MUWT(IPIK)).LE.0) GO TO 320
          IF (IFUN(J).EQ.IU(IPIK).AND.(IFUY(J).EQ.MYR.OR.IFUY(J).EQ.99.0
     +    R.(MYR.EQ.9.AND.IFUY(J).EQ.(NFIN-1)))) THEN
            DO 310 L=1,NSIT-1
              IF (IFUS(J).EQ.LABSI(L))  THEN
                INS=L
                IF(NSCAP(INS) .LE. 0)GO TO 295
                IF (MUREQ(IPIK).LE.(NSCAP(INS)-NSPOC(INS)))  THEN
                  LP = IPUN(IPIK)
                  CALL IALLUN (IPIK,INS)
                  JKP(LP)=JKP(LP)-MUREQ(IPIK)
                  WRITE (*,290) IFUN(J),MYR,IFUS(J)
 290              FORMAT (1X,'DESIGNATION OF UNIT',I5,' IN YR',I3,' TO',
     +' SITE ',A3)
                  IF(IPR19 .NE. 0)WRITE (19,290) IFUN(J),MYR,IFUS(J)
                  GO TO 330
                ENDIF
 295            WRITE (*,300) IFUN(J),MYR,IFUS(J)
 300            FORMAT (1X,'UNABLE TO DESIGNATE UNIT',I5,' IN YR',I3,' T
     +O',' SITE ',A3)
                IF(IPR19 .NE. 0)WRITE (19,300) IFUN(J),MYR,IFUS(J)
                GO TO 330
              ENDIF
 310        CONTINUE
            WRITE (*,300) IFUN(J),MYR,IFUS(J)
            IF(IPR19 .NE. 0)WRITE (19,300) IFUN(J),MYR,IFUS(J)
            GO TO 330
          ENDIF
 320    CONTINUE
 330  CONTINUE
```

```
C
C  IN THE 'REDUCE PROJECT DISPERSION' OPTION, FOR ALL YEARS, ATTEMPT
C  TO ALLOCATE, IN PROJECT PRIORITY ORDER, EACH PROJECT INTACT
C  (UNBROKEN) TO THE PROJECT'S GOAL SITE, THE PROJECT'S
C  IN-PLACE SITE, OR TO THE SITE CLOSEST TO THE PROJECT GDP.
C
C  INSURE THAT THESE PROJECT ALLOCATIONS ARE DONE EXACTLY ONCE PER
C  YEAR.  WHEN THE YEAR IS NOT THE GOAL YEAR ( NYR = 1 OR NFIN),
C  DO THESE ALLOCATIONS DURING THE 'MARK UNITS' CYCLE WHEN THE
C  FIRST SITE (ISIT = 1) IS BEING PROCESSED IN THAT CYCLE.
C
  340   IF (KPKG.LE.1)  THEN
        IF ((ISIT.EQ.1).OR. (NYR.EQ.1.OR.NYR.EQ.NFIN)) THEN
          DO 280 L=1,NST
            N=INDS(L)
            IF (IXO(N) .NE. 0)GO TO 280
            NZ = ICSTAY(N)
            NS = 0
C
C  IF PROJECT N CAN FIT AT ITS GOAL SITE (I.E. ITS FINAL YEAR ALLOCATION
C  SITE), THEN ALLOCATE IT THERE INTACT, IF POSSIBLE
C
            IF (ICSTAY(N) .GT. 0)THEN
              NS = ICSTAY(N)
              NLM = NUNP(N) + MINP(NS)
        IF (JKP(N).LE.(NSCAP(NS)-NSPOC(NS)).AND.NLM .LE. MXLIST)THEN
                IF (MYR.EQ.IBUGYR .OR. IBUGYR .EQ. 99)
     +          WRITE (19,265) N,LABSI(NS),NSCAP(NS),
     +          NSPOC(NS),MPREQ(N)
  265           FORMAT (' 1 $C P,NS,CA,SP,MPR=',I5,5X,A5,2I7,5X,I6)
                DO 244 KK=1,NUNP(N)
                  ID=IDPKG(N,KK)
                  IF(MOF(ID) .NE. 0)THEN
                    CALL IALLUN (ID,NS)
                    JKP(N)=JKP(N)-MUREQ(ID)
                  ENDIF
  244           CONTINUE
                GO TO 280
              ENDIF
            ENDIF
C
C  IF PROJECT N WAS ALLOCATED INTACT THE PREVIOUS YR, ALLOCATE IT
C  INTACT, IF POSSIBLE, TO ITS IN-PLACE SITE.
C
            IF(IPSTAY(N) .GT. 0)THEN
              ID = IDPKG(N,1)
              NS = IUSIT(ID)
              NLM = NUNP(N) + MINP(NS)
        IF (MPREQ(N).LE.(NSCAP(NS)-NSPOC(NS)).AND.NLM .LE. MXLIST)THEN
                IF (MYR.EQ.IBUGYR .OR. IBUGYR .EQ. 99)
     +          WRITE (19,242) N,LABSI(NS),NSCAP(NS),NSPOC(NS),MPREQ(N)
  242           FORMAT (' 1 $* P,NS,CA,SP,MPR=',I5,5X,A5,2I7,5X,I6)
                DO 245 KK=1,NUNP(N)
```

```
                ID=IDPKG(N,KK)
                IF(MOF(ID) .NE. 0)THEN
                  CALL IALLUN (ID,NS)
                  JKP(N)=JKP(N)-MUREQ(ID)
                ENDIF
245           CONTINUE
              GO TO 280
            ENDIF
          ENDIF
C
C  ATTEMPT TO ALLOCATE THE PROJECT INTACT (AS AN UNBROKEN PROJECT)
C  TO THE SITE NEAREST (ON AVERAGE) TO THAT PROJECT'S UNIT SET GDPS.
C
        NTO=NTOX(N)
          DO 270 I=1,NTO
            NS=ISTO(N,I)
            IF(NSCAP(NS) .LE. 0) GO TO 270
            NLM = NUNP(N) + MINP(NS)
      IF (MPREQ(N).LE.(NSCAP(NS)-NSPOC(NS)).AND.NLM .LE. MXLIST)THEN
C
C  NOTE THAT PROJECT N WAS ALLOCATED IN-PLACE THIS YR AT SITE NS. IF THIS
C  IS THE FIRST YR PROCESSED (GOAL YR) NOTE PROJECT N GOAL SITE NS
C
                IPSTAY(N) = NS
                IF(NYR .EQ. 1) ICSTAY(N) = NS
              IF (MYR.EQ.IBUGYR .OR. IBUGYR .EQ. 99)
    +         WRITE (19,250) N,LABSI(NS),NSCAP(NS),
    +         NSPOC(NS),MPREQ(N)
250           FORMAT (' 1 $$ P,NS,CA,SP,MPR=',I5,5X,A5,2I7,5X,I6)
              DO 260 KK=1,NUNP(N)
                ID=IDPKG(N,KK)
                IF(MOF(ID) .NE. 0)THEN
                  CALL IALLUN (ID,NS)
                  JKP(N)=JKP(N)-MUREQ(ID)
                ENDIF
260           CONTINUE
              GO TO 280
            ENDIF
270       CONTINUE
280     CONTINUE
      ENDIF
    ENDIF
C
C  FOLLOWING IS THE MAIN UNIT SET ALLOCATION LOOP, WHICH , DURING THE
C  ALLOCATION CYCLE, PROCESSES NORMAL UNIT SET ALLOCATIONS, USUALLY
C  ONE AT A TIME, IN THE ORDER SPECIFIED EARLIER.  IN THE ' REDUCE
C  PROJECT DISPERSION' OPTION,  MANY UNIT SETS IN A PROJECT MAY BE
C  ALLOCATED INSTEAD OF A UNIT.  THE ALLOCATED UNITS WILL THEN BE
C  NOT BE ALLOCATED WHEN PROCESSED IN SUBSEQUENT CYCLES OF THIS LOOP.
C  IF THIS CYCLE IS USED ONLY TO MARK UNIT SETS FOR RETENTION,
C  (ISIT .GT. 0) IT DOES NOT ALLOCATE NOW, BUT FUNCTIONS AS
C  A 'MARK UNIT SET' CYCLE TO ONLY MARK SELECTED UNIT SETS WHICH ARE
C  IN-PLACE AT SITE ISIT ISIT FOR ALLOCATION IN A LATER 'ALLOCATION CYCLE'
CALL TO THIS ROUTINE.
```

```
C
      DO 540 K=1,IL2
        IUN(K)=MINDS(K)
        IF ((MUREQ(IUN(K))+MUWT(IUN(K))).LE.0) GO TO 540
        IPIK=IUN(K)
        IZ=IU(IPIK)
        IF (MOF(IPIK).EQ.0) GO TO 540
C
C  IF THIS IS ONLY A 'MARK UNIT SET' CYCLE, THEN, IF THIS
C  UNIT SET IS IN-PLACE AT ITS GOAL SITE ISIT AND CAN FIT
C  AT SITE ISIT, THEN, IF FEASIBLE, MARK (DESIGNATE) THIS UNIT SET
C  FOR ALLOCATION TO (RETENTION AT) ITS GOAL SITE IN A FUTURE
C  'ALLOCATION CYCLE' CALL TO THIS ROUTINE.
C  DO NOT MARK A UNIT SET IF THERE WOULD BE NO ROOM BECAUSE THE SITE
C  WOULD BE ALREADY FILLED WITH MARKED UNIT SETS OF HIGHER PRIORITY.
C
        IF (ISIT.GT.0)  THEN
          NH=IHOLD(IZ)
          IF (IHOLD(IZ).NE.ISIT.AND.NSCAP(NH).GT.0) GO TO 540
          IF ((IRES(ISIT)+MUREQ(IPIK)).GT.(NSCAP(ISIT)-NSPOC(ISIT))) GO
     +    TO 540
          ISET(IPIK)=ISIT
          IRES(ISIT)=IRES(ISIT)+MUREQ(IPIK)
          GO TO 540
        ENDIF
        LP=IPUN(IPIK)
C
C FOLLOWING CODE CAN BE HANDY FOR DEBUGGING. REMOVE COMMENTS TO REACTIVATE
C
C     IF (NYR .GT. 1 .AND. NYR .LT. 5)THEN
C     NH = IHOLD(IZ)
C     WRITE(17,341)MYR,IZ,MOF(IPIK),LABSI(NH),IUSIT(IPIK),MINDS(K),K
C 341 FORMAT(1X,'YR,IZ,MO,IH,IUS,PR,K=',3I5,A4,I5,I10,I5)
C     WRITE(17,342)IZ,LABSI(IUSIT(IPIK)),JPRI(IPIK),NPRI(IPIK),
C    +ISET(IPIK)
C 342 FORMAT(1X,I5,' IP =',A4,3I10)
C     ENDIF
C
C  THE FOLLOWING IF ... THEN CLUSTER ALLOCATES UNIT SETS BY ATTEMPTING
C  TO REDUCE PROJECT DISPERSION WHILE KEEPING UNIT SETS AS CLOSE AS
C  POSSIBLE TO GDPS.
C
        IF (KPKG.LE.1)  THEN
          N=LP
C
C
C  IF THIS UNIT SET WAS MARKED EARLIER FOR ALLOCATION AT ITS IN-PLACE
C   GOAL SITE, THEN ALLOCATE IT NOW, IF POSSIBLE.  (IF NOT, OTHER
C   STATEMENTS WILL ALLOCATE IT LATER.) THESE UNIT SETS WILL HAVE BEEN
C   ORDERED SO AS TO BE PROCESSED FIRST IN THIS ALLOCATION LOOP.
C
          IF (ISET(IPIK).GT.0)  THEN
            INS=ISET(IPIK)
            IF ((NSCAP(INS)-NSPOC(INS)-MUREQ(IPIK)).GE.0
```

```
     +      .AND. NSCAP(INS) .GT. 0 )  THEN
              CALL IALLUN (IPIK,INS)
              JKP(LP)=JKP(LP)-MUREQ(IPIK)
              IF (MYR.EQ.IBUGYR .OR. IBUGYR .EQ. 99)
     +        WRITE (19,350) IU(IPIK),N,JKP(N),LABSI(INS),
     +        NSCAP(INS),NSPOC(INS),MUREQ(IPIK)
  350         FORMAT (' 1* U,P,J,NS,CA,SP,M=',2I5,I6,5X,A5,2I7,I6)
              GO TO 540
            ENDIF
          ENDIF
C
C  IF THIS IS NOT THE GOAL YEAR, AND IF THE GOAL SITE FOR THIS
C  UNIT SET HAS ROOM FOR IT, THEN ALLOCATE IT THERE NOW.
C  (IT WAS NOT THERE LAST YEAR, ELSE IT WOULD HAVE BEEN MARKED
C  FOR RETENTION AND ALREADY ALLOCATED.)
C
          IF (NYR.GT.1.AND.NYR.LT.NFIN)  THEN
            INS=IHOLD(IZ)
              IF (NSCAP(INS).GT.(NSPOC(INS)+MUREQ(IPIK)))  THEN
                  CALL IALLUN (IPIK,INS)
                  IF (MYR.EQ.IBUGYR .OR. IBUGYR .EQ. 99)
     +            WRITE (19,360) IU(IPIK),N,JKP(N),LABSI(INS),
     +            NSCAP(INS),NSPOC(INS),MUREQ(IPIK)
  360             FORMAT (' 1H* U,P,J,NS,CA,SP,M=',2I5,I6,5X,A5,2I7,I6)
                  JKP(N)=JKP(N)-MUREQ(IPIK)
                  GO TO 540
              ENDIF
          ENDIF
C
C  IF THIS IS NOT THE GOAL YEAR AND IF THERE IS ROOM FOR THE
C  UNIT SET TO STAY AT ITS IN-PLACE SITE, THEN ALLOCATE IT THERE.
C
          IP = IUSIT(IPIK)
          IF (NYR.GT.1.AND.NYR.LT.NFIN)  THEN
             IF (NSCAP(IP).GT.(NSPOC(IP)+MUREQ(IPIK)))  THEN
               CALL IALLUN (IPIK,IP)
               IF (MYR.EQ.IBUGYR .OR. IBUGYR .EQ. 99)
     +         WRITE (19,365) IU(IPIK),N,JKP(N),LABSI(IP),
     +         NSCAP(IP),NSPOC(IP),MUREQ(IPIK)
  365          FORMAT (' 1.5I* U,P,J,NS,CA,SP,M=',2I5,I6,5X,A5,2I7,I6)
               JKP(N)=JKP(N)-MUREQ(IPIK)
               GO TO 540
             ENDIF
          ENDIF
C
C  N IS THE PROJECT TO WHICH THIS UNIT SET BELONGS. ON THE SITE PREFERENCE
C  LIST FOR THIS PROJECT N, WHICH IS A LIST OF SITES ORDERED ACCORDING
C  TO THEIR AVERAGE CLOSENESS TO THE PROJECT N GDP, SELECT THE LEAST
C  NUMBER OF SITES, MOVING DOWN FROM THE BEGINNING OF THE LIST, WHICH
C  WHICH TOGETHER THEORETICALLY HAVE SUFFICIENT TOTAL UNOCCUPIED SPACE
C  FOR THE UNALLOCATED PORTION (JKP(N) OF THE PROJECT N.  HEREAFTER,
C  REFER TO THESE AS THE CURRENT SITE SHORTLIST FOR PROJECT N.
C
  375     NX=NUNP(N)
```

```
      NTO=NTOX(N)
      ISUM=0
      DO 380 I=1,NTO
        NKTOT=I
        NS=ISTO(N,I)
        ISUM=ISUM+NSCAP(NS)-NSPOC(NS)
        IF (ISUM.GT.JKP(N))  THEN
          NQ=MIN(NKTOT+1,NTO)
          NKPLUS=NQ
          GO TO 390
        ENDIF
380   CONTINUE
390   NST=0
      IZ=IU(IPIK)
      IF (IP.LT.1.OR.IP.GT.MXSITE) IP=NS
C
C  INDEX THE SITES IN THE SITE SHORTLIST ACCORDING TO THEIR CLOSENESS TO THIS
C  UNIT SET'S GDP. (WE WILL SOON SELECT THE BEST OF THESE SITES.)
C
      DO 410 L=1,NKTOT
        NS=ISTO(N,L)
        IF (NSCAP(NS).LE.(NSPOC(NS)+MUREQ(IPIK))) GO TO 410
        NST=NST+1
        IX(NST)=MUWT(IPIK)*(IDIST(NS,IPIK)-LDIS(IPIK))
        IF (MYR.EQ.IBUGYR .OR. IBUGYR .EQ. 99)
     +  WRITE (19,400) IU(IPIK),N,JKP(N),LABSI(NS),
     +  NSCAP(NS),NSPOC(NS),MUREQ(IPIK),IX(NST)
400     FORMAT (' 2* U,P,J,NS,CA,SP,M=',2I5,I6,5X,A5,2I7,I6,2X,I7)
        INDS(NST)=NS
410   CONTINUE
C
C  IF THE CURRENT 'SITE SHORTLIST' IS EMPTY, THEN, IN THE ORIGINAL
C  FULL SITE PREFERENCE LIST, SELECT THE ONE CLOSEST TO THIS UNIT SET'S
C  GDP WHICH ALSO HAS ROOM FOR IT. ALLOCATE THIS UNIT SET TO THAT SITE IF
C  POSSIBLE. IF NOT POSSIBLE, ALLOCATE IT TO THE OVERFLOW SITE.
C
      IF (NST.LE.0)  THEN
        IUS=0
        DO 440 J=NKPLUS,NTO
          NS=ISTO(N,J)
          IF (NSCAP(NS).GT.(NSPOC(NS)+MUREQ(IPIK)))  THEN
            CALL IALLUN (IPIK,NS)
            IF (MYR.EQ.IBUGYR .OR. IBUGYR .EQ. 99)
     +      WRITE (19,430) IU(IPIK),N,JKP(N),LABSI(NS),
     +      NSCAP(NS),NSPOC(NS),MUREQ(IPIK)
430         FORMAT (' 3M* U,P,J,NS,CA,SP,M=',2I5,I6,5X,A5,2I7,I6)
            JKP(N)=JKP(N)-MUREQ(IPIK)
            GO TO 540
          ENDIF
440     CONTINUE
        CALL IALLUN (IPIK,MSIT)
        JKP(N)=JKP(N)-MUREQ(IPIK)
C
C  FROM THE CURRENT 'SITE SHORTLIST',  CHOOSE THE SITE CLOSEST TO THIS UNIT
```

```
C  SET'S GDP AND ALLOCATE THIS UNIT SET THERE.
C
          ELSE
            CALL ORDER (NST)
            NL=INDS(1)
            IF (IUS.GT.0) NL=IUS
            IF ( (MYR.EQ.IBUGYR .OR. IBUGYR .EQ. 99) .AND.IUS.GT.0)
     +      WRITE (19,450) IU(IPIK),N,JKP(N),
     +      LABSI(NL),NSCAP(NL),NSPOC(NL),MUREQ(IPIK)
  450       FORMAT (' 4I* U,P,J,NS,CA,SP,M=',2I5,I6,5X,A5,2I7,I6)
            IF ( (MYR.EQ.IBUGYR .OR. IBUGYR .EQ. 99) .AND.IUS.LE.0)
     +      WRITE (19,460) IU(IPIK),N,JKP(N),
     +      LABSI(NL),NSCAP(NL),NSPOC(NL),MUREQ(IPIK)
  460       FORMAT (' 4M* U,P,J,NS,CA,SP,M=',2I5,I6,5X,A5,2I7,I6)
            CALL IALLUN (IPIK,NL)
            JKP(N)=JKP(N)-MUREQ(IPIK)
          ENDIF
          GO TO 540
        ENDIF
C
C  HERE ENDS THE CLUSTER WHICH ALLOCATES UNIT SETS BY ATTEMPTING
C  TO REDUCE PROJECT DISPERSION WHILE KEEPING UNIT SETS AS CLOSE AS
C  POSSIBLE TO GDPS.
C
C  THE FOLLOWING STATEMENTS (THROUGH STATEMENT 540) ATTEMPT TO ALLOCATE
C  EACH UNIT SET TO THE BEST AVAILABLE SITE WHILE IGNORING DISPERSION
C  OF PROJECTS.
C
C  IF THIS UNIT SET WAS MARKED FOR ALLOCATION TO ITS IN-PLACED GOAL
C  SITE, THEN ALLOCATE IT THERE NOW, IF POSSIBLE.  (IF NOT, OTHER
C  STATEMENTS WILL ALLOCATE IT LATER.)  THESE UNIT SETS HAVE
C  BEEN ORDERED SO AS TO BE PROCESSED FIRST IN THIS ALLOCATION LOOP.
C
       IF (ISET(IPIK).GT.0)  THEN
          INS=ISET(IPIK)
          IF ((NSCAP(INS)-NSPOC(INS)-MUREQ(IPIK)).GT.0)  THEN
            CALL IALLUN (IPIK,INS)
            IF (MYR.EQ.IBUGYR .OR. IBUGYR .EQ. 99)
     +      WRITE (19,470) MYR,IU(IPIK),LP,LABSI(INS)
     +      ,NSCAP(INS),NSPOC(INS),MUREQ(IPIK)
  470       FORMAT (' 1 YR,IU,LP,NS,CA,SP,MU=',3I5,5X,A5,2I7,I6)
            GO TO 540
          ENDIF
        ENDIF
        IZ=IU(IPIK)
        INS=IHOLD(IZ)
C
C  IF THIS IS NOT THE GOAL YEAR, AND IF THE GOAL SITE FOR THIS UNIT
C  SET HAS SUFFICIENT SPACE FOR IT, THEN ALLOCATE IT THERE NOW.
C  (IT WAS NOT THERE LAST YEAR, ELSE IT WOULD HAVE BEEN MARKED
C  FOR RETENTION AND ALREADY ALLOCATED.)
C
        IF (NYR.GT.1.AND.NYR.LT.NFIN)  THEN
          IF ((NSCAP(INS)-NSPOC(INS)-MUREQ(IPIK)).GT.0)  THEN
```

```
          CALL IALLUN (IPIK,INS)
         IF (MYR.EQ.IBUGYR .OR. IBUGYR .EQ. 99)
     +    WRITE (19,480) MYR,IU(IPIK),LP,LABSI(INS),
     +    NSCAP(INS),NSPOC(INS),MUREQ(IPIK)
 480      FORMAT (' 2 YR,IU,LP,NS,CA,SP,MU=',3I5,5X,A5,2I7,I6)
          GO TO 540
        ENDIF
C
C IF THIS IS NOT THE GOAL YEAR AND IF THERE IS SUFFICIENT SPACE FOR THE
C UNIT SET TO STAY AT ITS IN-PLACE SITE, THEN ALLOCATE IT THERE.
C
        INS=IUSIT(IPIK)
        IF ((NSCAP(INS)-NSPOC(INS)-MUREQ(IPIK)).GT.0)  THEN
          CALL IALLUN (IPIK,INS)
          IF (MYR.EQ.IBUGYR .OR. IBUGYR.EQ. 99)
     +    WRITE (19,490) MYR,IU(IPIK),LP,LABSI(INS)
     +    ,NSCAP(INS),NSPOC(INS),MUREQ(IPIK)
 490      FORMAT (' 3I YR,IU,LP,NS,CA,SP,MU=',3I5,5X,A5,2I7,I6)
          GO TO 540
        ENDIF
      ENDIF
C
C CHOOSE THE SITE CLOSEST TO THIS UNIT SET'S GDP WHICH HAS SPACE FOR
C THIS UNIT SET AND ALLOCATE THIS UNIT SET THERE, IF POSSIBLE. IF
C NOT POSSIBLE, ALLOCATE THIS UNIT SET TO THE OVERFLOW SITE..
C
 500  NST=0
      DO 520 NS=1,NSIT
        IF ((NSCAP(NS)-NSPOC(NS)-MUREQ(IPIK)).LE.0) GO TO 520
        NST=NST+1
        IX(NST)=MUWT(IPIK)*(IDIST(NS,IPIK)-LDIS(IPIK))
        IF ( (MYR.EQ.IBUGYR .OR. IBUGYR .EQ. 99) .AND.NSCAP(NS).GT.0)
     +  WRITE (19,510) MYR,IU(IPIK),
     +  LP,LABSI(NS),NSCAP(NS),NSPOC(NS),MUREQ(IPIK),IX(NST),IRES(NS)
 510    FORMAT (' 4 YR,IU,LP,NS,CA,SP,MU=',3I5,5X,A5,2I7,I6,3X,I10,
     +   I6)
        INDS(NST)=NS
 520  CONTINUE
      IF (NST.EQ.0)  THEN
        CALL IALLUN (IPIK,MSIT)
      ELSE
        CALL ORDER (NST)
        CALL IALLUN (IPIK,INDS(1))
        NL=INDS(1)
        IF (MYR.EQ.IBUGYR .OR. IBUGYR .EQ. 99)
     +  WRITE (19,530) MYR,IU(IPIK),LP,LABSI(NL),NSCAP(NL),
     +  NSPOC(NL),MUREQ(IPIK)
 530    FORMAT (' 5 YR,IU,LP,NS,CA,SP,MU=',3I5,5X,A5,2I7,I6)
      ENDIF
 540 CONTINUE
     RETURN
     END
```

```
$LARGE
      SUBROUTINE DISPAL (IG,MPC,MSPOC,IMAXP,XKSUM,ITW,ITS)
C
C  THIS ROUTINE COMPILES STATISTICS ON SUMMARIES OF ALLOCATIONS
C  PERFORMED.  IN THE ALLOCATION PHASE IT ALSO GENERATES, AS OUTPUT,
C  FILE 12.  THIS FILE 12 USED AS INPUT TO THE ASSESSMENT
C  PHASE OF THE MODEL.  IN THE ASSESSMENT PHASE, THIS ROUTINE OPTIONALLY
C  GENERATES FILE 20, WHICH IS ESSENTIALLY FILE 12 SORTED BY PROJECT
C  AND SUPPLEMENTED WITH ADDITIONAL OUTPUT INFORMATION.  ALSO, IN
C  THE ALLOCATION PHASE, THE GOAL SITE FOR EACH ALLOCATED
C  UNIT SET IS SET EQUAL TO THE ALLOCATION SITE.
C
$INCLUDE:'COMMON.INC'
$INCLUDE:'NEWCOM.INC'
      DIMENSION
     +    FILL(MXSITE),  ICUMA(MXSITE)
      REAL LK, ITW
      CHARACTER*3  ISI
      IF (IBUGYR.NE.0)  THEN
        WRITE (19,10) MYR,IG
   10   FORMAT (///,1X,' * YR =',I3,' ++  CURRENT ALLOCATION RESULTS',' 
     +FOR PKG ++',I3)
      ENDIF
      NITOT=0
      ZKDIST=0
      ITW=0
      ITS=0
      NUMP = 0
      XKSUM=0
      MPC=0
      MSPOC=-999
      IMAXP=-99
      DO 60 NS=1,MSIT
        IPC=0
        FILL(NS)=0.
        ICUMA(NS)=0
        IF(IND5 .NE. 0 .AND. NS .GE. NSIT)GO TO 60
        MM=MINP(NS)
        IF (MM.GT.0)  THEN
          DO 50 JK=1,MM
            XMOEU=-1.
            IF (MM .GT. MXLIST)THEN
              JFUL = 1
              IERR = 1
              WRITE(*,52)LABSI(NS),MYR,MXLIST
              IF(IPR19 .NE. 0)WRITE(19,52)LABSI(NS),MYR,MXLIST
   52         FORMAT(1X,'FATAL ERROR : NR UNITS ALLOCATED TO SITE',
     +        A4,' IN YR',I3,' EXCEEDS',I4)
            ENDIF
            IF (NALLU(NS,JK).EQ.0) GO TO 50
            IL=NALLU(NS,JK)
            IG = IPUN(IL)
            IF (MUWT(IL)+MUREQ(IL).LE.0)  THEN
```

```
C
C MARK UNITS THAT ARE NOT REALLY PRESENT
C
              IWTMV(IL)=0
              LK= - 10*MUWT(IL) - 15
              GO TO 20
            ENDIF
            IF (NS .GE. NSIT)IWTMV(IL) = 0
            IZ=IU(IL)
C
C SET THE GOAL SITE EQUAL TO THE ALLOCATION SITE FOR THE
C UNIT ALLOCATED.
C
            IF (NYR.EQ.1.AND.IND5 .GE. 0) IHOLD(IZ)=NS
            IPC=1
            LK=0
            IF (NS.NE.MSIT) ITW=ITW+MUWT(IL)
            NUMP = NUMP + 1
            ICUMA(NS)=ICUMA(NS)+MUREQ(IL)
            ITS=ITS+MUREQ(IL)
            ZZ=LDIS(IL)*MUWT(IL)
            IF (NSCAP(NS).GT.0.AND.NS.LT.NSIT)  THEN
              LK= IDIST(NS,IL)*MUWT(IL)
              XKSUM=XKSUM+NDIS(IL)
              ZKDIST=ZKDIST+LK
            ENDIF
   20       MTU=MUWT(IL)
            IF (MUWT(IL).LE.0) MTU=1
            IF (IUSIT(IL).LE.0.OR.IUSIT(IL).GT.MXSITE) IUSIT(IL)=MXSITE
            IF (IND5.NE.0)  THEN
C
C  FILE20 IS EXTREMELY USEFUL, BUT IS OPTIONAL
C
            IF (IREG.GT.0 .AND. IPR20 .NE. 0) THEN
            ISI = '000'
            IF (MYR .NE. 1 .AND. IUSIT(IL) .GT. 0)ISI = LABSI(IUSIT(IL))
            IF (MYR .EQ. 1 .AND. MUSIT(IL) .GT. 0)ISI = LABSI(MUSIT(IL))
              WRITE (20,30) MYR,IG,IU(IL),LABSI(NS),
     +        MUREQ(IL),MUWT(IL),ISI,IWTMV(IL)/10.,
     +        LK/10.,LABSI(IMIN(IL))
            ENDIF
   30         FORMAT (I3,I4,I4,1X,A3,2I8,1X,A3,2F10.0,1X,A3)
              GO TO 50
            ENDIF
            LV = (LK)/(10.*MTU)+.5
            IF (MYR.NE.9 .AND. IND5 .EQ. 0 .AND. IPR12 .NE. 0)THEN
              ISI = '000'
          IF (MYR .NE. 1 .AND. IUSIT(IL) .GT. 0)ISI = LABSI(IUSIT(IL))
            IF(MYR .EQ. 1 .AND. JUN(IL).GT. 0 )ISI = LABSI(JUN(IL))
              WRITE (12,40) MYR,IU(IL),LABSI(NS),LV,JPRI(IL),
     +        LABSI(IMIN(IL)),ISI
   40         FORMAT (I3,I4,A3,I4,I7,2A3)
            ENDIF
            NITOT=NITOT+1
```

```
 50       CONTINUE
        ENDIF
        MPC=MPC+IPC
        MP=NSCAP(NS)-NSPOC(NS)
        IF (NSPOC(NS).GT.0.AND.MP.GT.MSPOC) MSPOC=MP
        X=NSPOC(NS)
        Y=NSCAP(NS)
        IF (Y.GT..001) FILL(NS)=X/Y
 60   CONTINUE
      IF (NUMP.GT. 0)  THEN
        XKSUM=XKSUM/NUMP
        ZKDIST=ZKDIST/NUMP
      ENDIF
      XKSUM=ZKDIST
      IF (IBUGYR.NE.0)  THEN
        WRITE (19,90) NITOT
 90     FORMAT (/,1X,' TOTAL NR OF ALLOCATED UNIT SETS =',I5,/)
        WRITE (19,100) ITS
100     FORMAT (/,1X,' TOTAL SPACE RQMNT OF ALLOCATED UNIT SETS =',I10,/
     +  )
        WRITE (19,110) ITW
110     FORMAT (/,1X,' TOTAL WEIGHT OF ALLOCATED UNIT SETS =',F14.0,/)
      ENDIF
      IMAXP=-99
      DO 130 NS=1,MSIT
        IF (ICUMA(NS).GT.IMAXP) IMAXP=ICUMA(NS)
        IF (ICUMA(NS).GT.0)  THEN
          IF (IBUGYR.NE.0) WRITE (19,120) LABSI(NS),ICUMA(NS),MINP(NS)
120       FORMAT (/,2X,'SITE=',A5,3X,' TOT ALLOCATED =',I5,3X,'NR UNITS
     +ALLOC=',I3,/)
        ENDIF
130   CONTINUE
      IF ((IREG.GT.0.AND.IB.EQ.NRP).OR.IND5.EQ.0)  THEN
        NTC=0
        DO 180 NS=1,MSIT
          NTC=NTC+NSCAP(NS)
          IF (MYR.NE.9 .AND. (IND5 .EQ. 0 .OR. IND5 .EQ. -1)
     +    .AND. IPR17 .NE. 0)
     +    WRITE (14,170) MYR,LABSI(NS),NSPOC(NS),FILL(NS)
170       FORMAT (I5,A5,I7,F5.2)
180     CONTINUE
      ENDIF
      RETURN
      END
```

```
$LARGE
      SUBROUTINE IALLUN (ID,JSITE)
C
C  THIS ROUTINE ALLOCATES UNIT SET ID TO SITE ISIT AND UPDATES ALL
C  ASSOCIATED STATUS VARIABLES.
C
$INCLUDE:'COMMON.INC'
$INCLUDE:'NEWCOM.INC'
C
C  IF THE (OVERFLOW) SITE (SITE -99) ALREADY HAS THE MAX (MXLIST)
C  NR OF UNITS ALLOCATED TO IT, THEN SHIFT THIS UNIT SET
C  ALLOCATION TO THE CONUS SITE (SITE -01), IF POSSIBLE.
C
      JU = IUSIT(ID)
      LP = IPUN(ID)
      IF (JSITE .NE. JU .AND. LLMURQ(ID) .GT. 0)NUCP(LP) = NUCP(LP) + 1
      IF (MYR .EQ. 1 .AND. JSITE .NE. JUN(ID))NUCP(LP) = NUCP(LP) + 1
      ISIT = JSITE
      IF (IFUL(ISIT) .NE. 0)THEN
        JFUL = 1
        IF(IPR19 .NE. 0)WRITE(19,5)MYR,LABSI(ISIT),MXLIST,ID
    5   FORMAT(1X,'ERROR - YR',I3,' NR ALLOC TO SITE',A4,
     +  ' EXCEEDS',I4,' - WITH UNIT',I5)
        ISIT = NSIT
      ENDIF
C
C  IF THE CONUS SITE (SITE -01) ALREADY HAS THE MAX (MXLIST) NR OF UNITS
C  ALLOCATED TO IT, THEN ABORT THE RUN.
C
      IF (MINP(NSIT) .GE. MXLIST)THEN
          IERR = 1
          IF(IPR19 .NE. 0)WRITE(19,6)MYR,MXLIST
    6     FORMAT(1X,' * FATAL ERROR - YR',I3,' NR ALLOC TO ',
     +    ' OVERFLOW SITES EXCEEDS',I4)
          RETURN
        ENDIF
      MINP(ISIT)=MINP(ISIT)+1
C
C  IF SITE ISIT WILL HAVE THE MAX (MXLIST) NR OF UNITS ALLOCATED
C  TO IT NOW, THEN CLOSE THIS SITE.
C
      IF (MINP(ISIT) .GE. MXLIST)THEN
        IFUL(ISIT) = 1
        NSCAP(ISIT) = -NSCAP(ISIT)
        WRITE(*,8)MYR,LABSI(ISIT),MXLIST
        IF(IPR19 .NE. 0)WRITE(19,8)MYR,LABSI(ISIT),MXLIST
    8   FORMAT(1X,' --WARNING : YR',I3,' SITE',A4,' AT LIMIT OF',I4,
     +  ' UNITS ALLOCATED-SITE NOW CLOSED')
      ENDIF
      JK=MINP(ISIT)
      NALLU(ISIT,JK)=ID
      NSPOC(ISIT)=NSPOC(ISIT)+MUREQ(ID)
```

```
      JS=IUSIT(ID)
      MOF(ID)=0
      IF (JS.LT.1.OR.JS.GT.NSIT)  THEN
        IWTMV(ID)=0
        NDIS(ID)=0
        RETURN
      ELSE
        IWTMV(ID)=MUWT(ID)*JDIST(JS,ISIT)
        NDIS(ID)=(IDIST(ISIT,ID)-LDIS(ID))*MUWT(ID)
        RETURN
      ENDIF
      END
```

```
$LARGE
      SUBROUTINE ORDER (N)
C
C  THIS ROUTINE REORDERS ARRAY IX(K) IN INCREASING ORDER. IT ALSO ORDERS
C  ARRAY INDS(K) IN THAT SAME ORDERING.
C
$INCLUDE:'COMMON.INC'
      DO 20 K=1,N-1
        ITEMP=IX(K)
        IND=K
        DO 10 L=K+1,N
          IF (IX(L).GT.ITEMP) GO TO 10
          ITEMP=IX(L)
          IND=L
   10   CONTINUE
        IXX=IX(K)
        IX(K)=ITEMP
        IX(IND)=IXX
        IXX=INDS(K)
        INDS(K)=INDS(IND)
        INDS(IND)=IXX
   20 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE SET (NSLIM,IND1,IPAS)
C
C  THIS ROUTINE ALLOWS A USER TO INTERACTIVELY SET INPUT PARAMETERS
C   NSLIM, KPKG, AND IPAS.  IT ALSO FIXES DEFAULT IND1 = 1.
C
$INCLUDE:'COMMON.INC'
      INTEGER
     +    SAO
      NFIN=8
      NSLIM=99
      IND1=1
      IPAS=1
      KPKG=1
      WRITE(*,8)
    8 FORMAT(/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,
     + ' **  1. LOAD DATA FOR THIS CASE INTO DRIVE **',/,
     +' **  2. AFTER DATA IS LOADED, PUSH ENTER KEY **')
      PAUSE
      OPEN (19,FILE='FILE19.TXT')
      WRITE (*,10)
   10 FORMAT (/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,
     +' ** TO ABORT THIS RUN AT ANY TIME, PUSH [CTRL-C] **',
     +//)
   20 INC=0
   30 WRITE (*,40)
   40 FORMAT (/,1X,'CHOOSE ONE OPTION',/,' 1. AVAILABLE STORAGE CASE ',
     +'(ENTRY = 1)',/,/,' 2. UNCONSTRAINED STORAGE CASE (ENTRY = 2)',/,
     +/,' 3. BASELINE CASE  (ENTRY = 3)',//)
   43 READ (*,*,ERR=120) SAO
      IF (SAO.EQ.1.OR.SAO.EQ.2.OR. SAO.EQ.3)  THEN
        IF (SAO.EQ.1)  THEN
          IPAS=1
          WRITE (*,50)
   50     FORMAT (/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,
     +' THIS IS AVAILABLE STORAGE CASE (CAN REDEFINE',
     +' LATER.)')
        ENDIF
        IF (SAO.EQ.2)  THEN
          IPAS=2
          WRITE (*,60)
   60     FORMAT (/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,
     +' THIS IS UNCONSTRAINED STORAGE CASE (CAN REDEFINE',
     +' LATER)')
        ENDIF
        IF (SAO.EQ.3)  THEN
          IPAS=3
          GO TO 350
        ENDIF
   70   WRITE (*,80)
   80   FORMAT (/,/,' ENTRY = 0 TO CONTINUE',/,/,' ENTRY = 9 TO SKIP REM
     +AINING INPUT OPTIONS',/,/)
        READ (*,*,ERR=90) IAO
        IF (IAO.EQ.0.OR.IAO.EQ.9)  THEN
```

```
          IF (IAO.EQ.9) GO TO 350
          IF (IAO.EQ.0) GO TO 140
        ENDIF
 90     JNC=JNC+1
      IF (JNC .LT. 5)THEN
         WRITE (*,100) JNC
100     FORMAT (/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,
     + ' ENTRIES OTHER THAN 0 OR 9 ARE INVALID',/'. TRY AGAIN',
     + ' (DEFAULT SET AFTER 5 ATTEMPTS). - TRIES =',I2,//)
         GO TO 70
      ELSE
          IAO=0
          GO TO 350
        ENDIF
      ENDIF
120   INC=INC+1
      IF (INC .LT. 5) THEN
        WRITE (*,130) INC
130   FORMAT (/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,
     +' ENTRIES OTHER THAN 1,2, OR 3 ARE INVALID',/'. TRY AGAI',
     +'N (DEFAULT SET AFTER 5 ATTEMPTS). - TRIES ='I2,//)
         WRITE(*,41)
 41   FORMAT (/,1X,'CHOOSE ONE OPTION',/,' 1. AVAILABLE STORAGE CASE ',
     +'(ENTRY = 1)',/,/,' 2. UNCONSTRAINED STORAGE CASE (ENTRY = 2)',/,
     +/,' 3. BASELINE CASE  (ENTRY = 3)',//)
        GO TO 43
      ELSE
        IPAS=1
        WRITE (*,50)
        INC = 0
        WRITE(*,161)
        GO TO 163
      ENDIF
140   INC=0
150   WRITE (*,160)
160   FORMAT (/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,
     + 1x,' CHOOSE ONE OPTION',/,
     + 1X,' 1. REDUCE PROJECT DISPERSION OVER',' SITES  (ENTRY',
     +'= 1)',/,/,'  2. IGNORE PROJECT DISPERSION  (ENTRY = 2)',//)
163   READ (*,*,ERR=210) SAO
      IF (SAO.EQ.1.OR.SAO.EQ.2)  THEN
        IF (SAO.EQ.1)  THEN
          KPKG=1
          WRITE (*,170)
170       FORMAT (/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,
     +' REDUCE PROJECT DISPERSION (CAN REDEFINE',' LATER. )')
        ENDIF
        IF (SAO.EQ.2)  THEN
          KPKG=2
          WRITE (*,180)
180       FORMAT (/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,
     +' IGNORE PROJECT DISPERSION (CAN REDEFINE LATER.)')
        ENDIF
190     WRITE (*,80)
```

```
      READ (*,*,ERR=200) IAO
      IF (IAO.EQ.0.OR.IAO.EQ.9)  THEN
        IF (IAO.EQ.9) GO TO 350
        IF (IAO.EQ.0) GO TO 220
      ENDIF
200   JNC=JNC+1
      WRITE (*,100) JNC
      IF (JNC.GE.5)  THEN
        IAO=0
        GO TO 350
      ENDIF
      GO TO 190
    ENDIF
210 INC=INC+1
    IF (INC .LT. 5)THEN
      WRITE (*,205) INC
205 FORMAT (/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,
   +' ENTRIES OTHER THAN 1 OR 2 ARE INVALID',/'. TRY AGAI',
   +'N (DEFAULT SET AFTER 5 ATTEMPTS). - TRIES ='I2,//)
      WRITE (*,161)
161 FORMAT (/,/,' CHOOSE ONE OPTION',/,
   + 1X,' 1. REDUCE PROJECT DISPERSION OVER',' SITES  (',
   +'ENTRY = 1)',/,/,'  2. IGNORE PROJECT DISPERSION  (ENTRY = 2)',//)
     GO TO 163
    ELSE
      KPKG=1
      WRITE (*,170)
      INC = 0
      WRITE(*,241)
      GO TO 243
    ENDIF
220 INC=0
    IF (IPAS .EQ. 2)GO TO 350
230 WRITE (*,240)
240 FORMAT (/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,
   +1X,'HOW MANY REDUNDANT SITES(BEYOND THE ESTIMATED MINIM'
   +,'UM',/' NEEDED EACH YEAR TO STORE ALL UNIT SETS) DO YOU ALLOW?',/
   +,'(ENTRY = NUMBER ALLOWED)',/,' (ENTRY = 99 ALLOWS ALL SITES)',//)
243 READ (*,*,ERR=280) SAO
    IF (SAO.GE.0.AND.SAO.LE.99)  THEN
      NSLIM=SAO
      WRITE (*,250) SAO
250   FORMAT (/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,
   +' NR ESTIMATED REDUNDANT SITES ALLOWED =',I2,' (CAN RE',
   +'DEFINE LATER)',/)
      JNC=0
260   WRITE (*,80)
      READ (*,*,ERR=270) IAO
      IF (IAO.EQ.0.OR.IAO.EQ.9) GO TO 350
270   JNC=JNC+1
      WRITE (*,100) JNC
      IF (JNC.GE.5)  THEN
        IAO=0
        GO TO 350
```

```
          ENDIF
          GO TO 260
        ENDIF
C
  280 INC=INC+1
        IF (INC .LT. 5)THEN
          WRITE (*,290) INC
  290 FORMAT (/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,
     +' ENTRIES OTHER THAN 0 THRU 99 ARE INVALID',/'. TRY AGAI',
     +'N (DEFAULT SET AFTER 5 ATTEMPTS). - TRIES =',I2,/)
          WRITE(*,241)
  241 FORMAT (/,/,1X,'HOW MANY REDUNDANT SITES(BEYOND',
     +' THE ESTIMATED MINIMUM',
     +/,' NEEDED EACH YEAR TO STORE ALL UNIT SETS) DO YOU ALLOW?',/,
     +'(ENTRY = NUMBER ALLOWED)',/,' (ENTRY = 99 ALLOWS ALL SITES)',//)
            GO TO 243
        ELSE
          NSLIM=99
          GO TO 350
        ENDIF
C
  350 IF (IPAS.EQ.1)THEN
          WRITE (*,360)
          WRITE(19,360)
        ENDIF
  360 FORMAT (/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,
     +' FOLLOWING ARE ITEM SETTINGS:',/,/,' (ITEM NR = 1):',
     +' THIS IS AVAILABLE STORAGE CASE')
        IF (IPAS.EQ.2)THEN
          WRITE (*,370)
          WRITE(19,370)
        ENDIF
  370 FORMAT (/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,
     +' FOLLOWING ARE ITEM SETTINGS:',/,/,' (ITEM NR = 1):',
     +' THIS IS UNCONSTRAINED STORAGE CASE')
        IF (IPAS.EQ.3)THEN
          WRITE (*,371)
          WRITE(19,371)
        ENDIF
  371 FORMAT (/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,
     +' FOLLOWING ARE ITEM SETTINGS:',/,/,' (ITEM NR = 1):',
     +' THIS IS BASELINE CASE')
        IF (IPAS.LE.2.AND.KPKG.EQ.1)THEN
          WRITE (*,380)
          WRITE(19,380)
        ENDIF
  380 FORMAT (' (ITEM NR = 2): REDUCE PROJECT DISPERSION')
        IF (IPAS.LE.2.AND.KPKG.EQ.2)THEN
          WRITE (*,390)
          WRITE(19,390)
        ENDIF
  390 FORMAT (' (ITEM NR = 2): IGNORE PROJECT DISPERSION.')
        IF (IPAS.EQ.1)THEN
          WRITE (*,400) NSLIM
```

```
      WRITE(19,400) NSLIM
    ENDIF
400 FORMAT (' (ITEM NR = 3): NR OF EST. REDUNDANT SITES ALLOWED =',I3)
    INC=0
410 WRITE (*,420)
420 FORMAT (/,/,' SET ENTRY = ITEM NR TO REDEFINE ASSOCIATED INPUT',/,
   +' SET ENTRY = 0 TO BEGIN RUN',//)
    READ (*,*,ERR=430) SAO
    IF (SAO.GE.0.AND.SAO.LE.3)  THEN
      IF (SAO.EQ.0) GO TO 450
      IF (SAO.EQ.1) GO TO 20
      IF (SAO.EQ.2 .AND. IPAS.LE.2) GO TO 140
      IF (SAO.EQ.3 .AND. IPAS.EQ. 1) GO TO 220
    ENDIF
430 INC=INC+1
    WRITE (*,440) INC
440 FORMAT (/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,/,
   +' ENTRIES OTHER THAN 0 THRU 3 ARE INVALID',/'. TRY AGAIN',
   +' (DEFAULT SET AFTER 5 ATTEMPTS). - TRIES =,'I2,/)
    IF (INC.GE.5)  THEN
      GO TO 450
    ENDIF
    GO TO 410
450 RETURN
    END
```

```
      FUNCTION CONVRT (DLAU1,DLOU1,DLAU,DLOU)
C**** DOUBLE PRECISION DEGLAT,DEGLNG,DLAU,DLOU,DLAU1,DLOU1,X,Y
C
C  USING THE ASSUMPTION OF A SPHERICAL EARTH, THIS ROUTINE
C  FINDS THE EXACT DISTANCE, IN KM, BETWEEN ANY TWO LOCATIONS IN THE
C  NORTHERN HEMISPHERE, GIVEN THAT THEY ARE EXPRESSED AS RADIANS OF
C  (1) LATITUDE AND (2) LONGITUDE EAST OF GREENWICH.
C
C THE ROUTINE OPERATES AS FOLLOWS : THE LINES OF LATITUDE AND LONGITUDE
C DRAWN THROUGH THE TWO LOCATIONS FORMS A POLYGON WITH THE LOCATIONS
C AT TWO "CORNERS".  IF  CHORDS ARE DRAWN CONNECTING THESE "CORNERS",
C AN ISOSCELES TRAPAZOID IS FORMED. (THE CHORDS ARE BENEATH THE EARTH,S
C SURFACE, EXCEPT FOR THE CORNERS.) SIMPLE PLANE GEOMETRY FINDS THE
C LENGTHS OF THE TOP, BOTTOM, AND SIDE OF THIS TRAPEZOID, AND USES THESE
C TO FIND THE LENGTH OF THE DIAGONAL CONNECTING THE TWO LOCATIONS. THIS
C DIAGONAL IS THEN PROJECTED BACK ONTO THE EARTH'S SURFACE AS THE
C DISTANCE BETWEEN THE TWO LOCATIONS.
C
C  ARGUMENTS :
C
C DLAU1       THE LATITUDE OF THE FIRST LOCATION, EXPRESSED AS RADIANS
C             NORTH OF THE EQUATOR.
C
C DLOU1       THE LONGITUDE OF THE FIRST LOCATION, EXPRESSED AS RADIANS
C             EAST OF THE GREENWICH MERIDIAN.  FOR POINTS WITH WEST
C             LONGITUDE, THE DEGREES LONGITUDE MUST BE SUBTRACTED FROM 360
C             BEFORE CONVERSION TO RADIANS.
C
C DLAU        THE RADIANS LATITUDE OF THE SECOND LOCATION.
C
C DLOU        THE RADIANS EAST LONGITUDE OF THE SECOND LOCATION.
C
C CONVRT      THE DISTANCE, IN KM, BETWEEN THE TWO LOCATIONS.
C
      IF (DLAU1.LT.DLAU)  THEN
        TEMP=DLAU
        DLAU=DLAU1
        DLAU1=TEMP
        TEMP=DLOU
        DLOU=DLOU1
        DLOU1=TEMP
      ENDIF
      DLON=ABS(DLOU-DLOU1)
      IF (DLON.GT.180) DLON=360-DLON
      TOP=2.*COS(DLAU1)*6378.388*SIN(DLON/2)
      BASE=2.*COS(DLAU)*6378.388*SIN(DLON/2)
      TOP=ABS(TOP)
      BASE=ABS(BASE)
      DLAT=DLAU1-DLAU
      SIDE=2.*6378.388*SIN(DLAT/2)
      X=(BASE-TOP)/2.
      Y=SQRT(SIDE**2-X**2)
      X=(BASE+TOP)/2.
```

```
CONVRT=SQRT(X**2+Y**2)
XX=2.*6378.388
CONVRT=XX*ASIN(CONVRT/XX)
RETURN
END
```

COMMON STATEMENT

```
      PARAMETER (MXFL=50)
      PARAMETER (MXUNIT=1000)
      PARAMETER (MXPKG=25)
      PARAMETER (MXSITE=30)
      PARAMETER (MXLIST=200)
      PARAMETER (MXHOLD=2000)
      COMMON
     +    IB,                          IBUGYR,
     +                                 IDIST(MXSITE,MXUNIT),
     +    IDPKG(MXPKG,MXLIST),
     +    IFPN(MXFL),                  IFPY(MXFL),
     +    IFPS(MXFL),                  IDUN,
     +    IDUNP,                       IFUN(MXFL),
     +    IFUY(MXFL),                  IFUS(MXFL),
     +    IHOLD(MXHOLD),               IMIN(MXUNIT),
     +    IERR,                        INDS(MXUNIT),
     +                                 INPLU(MXLIST,MXSITE),
     +                                 IPR19D,
     +    IPRIO(MXPKG),                IPUN(MXUNIT),
     +    IU(MXUNIT),                  IREG,
     +    IRES(MXSITE),                ISET(MXUNIT),
     +    ISTO(MXPKG,MXSITE),          ITOT(MXPKG),
     +    IUSIT(MXUNIT),               IWTMV(MXUNIT),
     +    IX(MXUNIT),                  JDIST(MXSITE,MXSITE),
     +    MUSIT(MXUNIT),               ICSTAY(MXPKG),
     +    IPSTAY(MXPKG),               JFUL,
     +    IFUL(MXSITE),                LLMURQ(MXUNIT)
      COMMON /SEG/
     +                                 LABSI(MXSITE),
     +    LDIS(MXUNIT),                MINP(MXSITE),
     +    MOF(MXUNIT),                 JKP(MXPKG),
     +    KPKG,                        JPRI(MXUNIT),
     +    MFL,                         NFIN,
     +    IPR20,                       IPR12,
     +    IPR19,                       IPR17
      COMMON /BIG/
     +    MPREQ(MXPKG),                MSIT,
     +    NUCP(MXPKG),                 NUMP,
     +    MUREQ(MXUNIT),               MUWT(MXUNIT),
     +    MYR,                         NALLU(MXSITE,MXLIST),
     +    NDIS(MXUNIT),                NIND(MXUNIT),
     +    NPAK,
     +    NPRI(MXUNIT),                NRP,
     +    NSCAP(MXSITE),               NSIT,
     +    NSPOC(MXSITE),               NTUN,
     +    NUIS(MXSITE),                NUNP(MXPKG),
     +    NTOX(MXPKG),                 NYR,
     +    ZNUC(8,50)
      CHARACTER*3
     +    IFPS,     IFUS,     LABSI
      REAL IDIST, LDIS, NDIS, IWTMV
```

```
      COMMON /NEW/
     +    IMALL,               IMTOT,
     +    IND5,                IPR(MXPKG),
     +    JSIT(MXUNIT),        JUN(MXUNIT),        MNUN,
     +    LMUREQ(MXHOLD),      LTOT,               MALL,
     +                         MPCT,               MTOT(MXPKG),
     +                         NCNT(9),            NB,
     +                         XDG,                ZTP
      REAL IMTOT
      CHARACTER*3
     +    JSIT
```

```
C   COMMON VARIABLES
C
C
C   IB                   INDEX (IN DO LOOP) OF PROJECT BEING PROCESSED
C
C   IBUGYR               WRITE FLAG : IF IBUGYR .NE. 0, LOTS OF DEBUG TABLES
                         ARE
C                        PRINTED TO ASSIST THE MODELER IN ASSESSING PROPER

C                        ALGORITHM OPERATION. (THIS IS ALWAYS SET = 0 BY
C                        DEFAULT, BUT CAN BE CHANGED VIA FILE15.TXT.
C
C   ICSTAY(IG)           THE NUMERIC SITE ID OF THE SITE (IF ANY) AT WHICH
C                        PROJECT IG WAS ALLOCATED INTACT IN THE
C                        GOAL YEAR.
C
C   IDIST(NS,N)          DISTANCE, IN KM, BETWEEN THE SITE WITH INDEX NS
C                        AND THE GDP LOCATION OF THE UNIT SET WITH INDEX N.
C
C   IDPKG(IG,N)           INDEX (AS INDEXED BY SITING) OF UNIT SET THAT IS THE
C                         N-TH UNIT SET (AS INDEXED BY SITING) IN PROJECT IG
C
C   IDUN                  CONTROL PARAMETER INSURING THAT ALLOCATIONS ON THE
C                         DESIGNATED UNIT ALLOCATIONS LIST ARE ONLY DONE
C                         ONCE (DURING THE 'MARK UNITS' CYCLE OF BYUNIT).
C                         IDUN = 0 MEANS ALLOCATIONS ARE NOT COMPLETED.
C                         IDUN = 1 MEANS THEY HAVE BEEN DONE ALREADY.
C
C   IDUNP                 CONTROL PARAMETER INSURING THAT ALLOCATIONS ON THE
C                         DESIGNATED PROJECT ALLOCATIONS LIST ARE ONLY DONE
C                         ONCE (DURING THE 'MARK UNITS' CYCLE OF BYUNIT).
C                         IDUNP = 0 MEANS ALLOCATIONS ARE NOT COMPLETED.
C                         IDUNP = 1 MEANS THEY HAVE BEEN DONE ALREADY.
C
C   IERR                  ERROR INDICATOR : IT IS SET = 1 WHEN A FATAL ERROR IS
C                         DETECTED.  IERR = 1 ABORTS THE RUN.
C
C
C   IFPN(K)               THE DESIGNATED PROJECT ID FOR THE K-TH INPUT ITEM
C                         ON THE DESIGNATED PROJECT ALLOCATIONS LIST (INPUT
C                         FILE7.TXT)
C
C   IFPS(K)               THE DESIGNATED SITE FOR THE K-TH INPUT ITEM
C                         ON THE DESIGNATED PROJECT ALLOCATIONS LIST (INPUT
C                         FILE7.TXT)
C
C   IFPY(K)               THE DESIGNATED YEAR FOR THE K-TH INPUT ITEM
C                         ON THE DESIGNATED PROJECT ALLOCATIONS LIST (INPUT
C                         FILE7.TXT)
C
C   IFUL(NS)              INDICATOR : SET = 1 WHEN SITE NS IS FILL (I.E. HAS
C                         MXLIST UNIT SETS ALLOCATED TO IT) AND IS
C                         SUBSEQUENTLY CLOSED (SITE NS CAPACITY IS THEN SET =
                          0)
```

```
C
C   IFUN(K)            THE DESIGNATED UNIT SET ID FOR THE K-TH INPUT ITEM
C                      ON THE DESIGNATED UNIT SET ALLOCATIONS LIST (INPUT
C                      FILE7.TXT)
C
C   IFUS(K)            THE DESIGNATED SITE FOR THE K-TH INPUT ITEM
C                      ON THE DESIGNATED UNIT SET ALLOCATIONS LIST (INPUT
C                      FILE7.TXT)
C
C   IFUY(K)            THE DESIGNATED YEAR FOR THE K-TH INPUT ITEM
C                      ON THE DESIGNATED UNIT SET ALLOCATIONS LIST (INPUT
C                      FILE7.TXT)
C
C   IHOLD(IZ)          THE INDEX OF THE GOAL SITE FOR THE UNIT SET
C                      WITH NUMERIC UNIT SET ID = IZ
C
C   IMALL              TOTAL AREA OF ALL UNIT SETS (OVER ALL PROJECTS)
C                      THAT WERE ALLOCATED BY THE ALGORITHM
C
C   IMIN(N)            INDEX OF NEAREST SITE TO THE GDP OF THE UNIT SET
                       N  C
C                      WITH INDEX N
C
C   IMTOT              TOTAL TONS (OVER ALL UNIT SETS) THAT ARE ALLOCATED
C
C   IND5               CONTROL PARAMETER : IND5 .EQ. 0 MEANS THAT THIS
C                      IS THE ALLOCATION PHASE. IND5 .EQ. 1 MEANS THAT
C                      THIS IS THE POST-ALLOCATION ASSESSMENT PHASE.
C                      IND5  = -1 MEANS THAT THIS IS THE BASELINE CASE,
C                      I.E. THE ASSESSMENT OF IN-PLACE IN ONLY THE FIRST YR.
C
C   INDS(K)            ARRAY ORDERED IN SUBROUTINE ORDER IN SAME INCREASING
C                      ORDER AS ARRAY IX(K).  USUALLY, INDS( ) CONTAINS
C                      THE INDEXES OF THE ORDERED IX ( ) VALUES .  THUS,
C                      E.G., IF IX IS SET EQUAL TO ELEMENTS FROM AN
C                      ARRAY ZZ( ), THEN, AFTER SUBROUTINE ORDER HAS
                       OPERATED
C                      ,IX(1) GIVES THE SMALLEST VALUE OF THE ZZ( ) AND
C                      ZZ(IND(1)) IS THAT SMALLEST VALUE.
C
C
C   INPLU(JK,NS)       UNIT SET INDEX OF JK-TH UNIT SET IN THE IN-PLACE LIST
C                      AT SITE NS FOR ONLY THOSE SETS IN THE PROJECT BEING

C                      PROCESSED.
C
C   IPR(IG)            ARRAY OF PROJECT IDs INDEXED ACCORDING TO INCREASING
C                      ORDER OF NUMERIC PROJECT PRIORITY. (E.G. IPR(1) IS
C                      THE ID OF THE HIGHEST PRIORITY PROJECT, I.E. THE
C                      ONE  WITH SMALLEST NUMERIC PROJECT PRIORITY.)
C
C   IPR12              WRITE FLAG : IPR12 .NE. 0 CAUSES WRITING OF
C                      ALLOCATION SUMMARIES TO FILE12.TXT.  IPR12 IS
C                      ALWAYS SET = 1 BY DEFAULT, BUT CAN BE
```

```
C                               CHANGED VIA INPUT FILE15.TXT
C
C    IPR17                      WRITE FLAG : IPR17 .NE. 0 CAUSES WRITING OF
C                               SITE FILL REPORT TO FILE14.TXT AND OF
C                               MOE SUMMARIES TO FILE17.TXT.  IPR17 IS
C                               ALWAYS SET = 0 BY DEFAULT, BUT CAN BE
C                               CHANGED VIA INPUT FILE15.TXT
C
C    IPR19                      WRITE FLAG : IPR19 .NE. 0 CAUSES WRITING OF
C                               ERROR MESSAGES TO FILE19.TXT.  IPR19 IS
C                               ALWAYS SET = 0 BY DEFAULT, BUT CAN BE
C                               CHANGED VIA INPUT FILE15.TXT
C
C    IPR20                      WRITE FLAG : IPR20 .NE. 0 CAUSES WRITING OF
C                               ALLOCATION SUMMARIES TO FILE19.TXT.  IPR20 IS
C                               ALWAYS SET = 0 BY DEFAULT, BUT CAN BE
C                               CHANGED VIA INPUT FILE15.TXT
C
C    IPRIO(IG)                  INPUT PRIORITY OF PROJECT IG ( 1 = HIGHEST)
C
C    IPSTAY(IG)                THE INDEX OF THE SITE (IF ANY) AT WHICH
C                              PROJECT IG WAS ALLOCATED INTACT IN THE
C                              PREVIOUS YEAR.
C
C    IPUN(N)                    INDEX OF PROJECT CONTAINING UNIT SET N
C
C    IREG                       CONTROL PARAMETER USED TO ENSURE THAT FILE14.TXT
C                               AND FILE20.TXT ARE WRITTEN ONLY DURING THE
C                               ALLOCATION PHASE
C
C    IRES(NS)                   AMOUNT OF UNOCCUPIED SPACE AT SITE NS WHICH IS

C                               DESIGNATED AS RESERVED FOR LATER ALLOCATIONS
C                               (IN THE 'ALLOCATION CYCLE' OF ROUTINE BYUNIT)
C                               OF 'MARKED' (IN A 'MARK UNITS' CYCLE OF BYUNIT)
C                               UNIT SETS TO SITE NS. THESE 'MARKED' UNIT SETS ARE
C                               THOSE WHICH WERE IN-PLACE AT , AND CAN FIT AT, THEIR
C                               GOAL SITES AND ARE THERFORE BEING ALLOCATED THERE.
C
C    ISET(N)                    INDEX OF THE SITE TO WHICH A UNIT SET WITH
C                               INDEX N IS MARKED FOR ALLOCATION IN A 'MARK UNITS'
C                               CYCLE. (IF MARKED, UNIT SET N IS INPLACE AT ITS GOAL
C                               SITE AND CAN FIT THERE.)
C
C    ISTO(IG,NS)                THE SITE ID OF THE NS-TH RANKED SITE IN TERMS
C                               OF INCREASING CLOSENESS TO THE PROJECT GDP FOR
C                               PROJECT IG.  (E.G. ISTO(IG,1) IS THE SITE ID OF
C                               THE CLOSEST SITE TO THE PROJECT IG GDP))
C
C    ITOT(IG)                   TOTAL WEIGHT OF ALL UNIT SETS IN PROJECT IG IN
C                               THE YEAR BEING PROCESSED
C
C    IU(N)                      UNIT SET NUMERIC ID OF THE UNIT SET WITH INDEX N
C                               IN THE YEAR BEING PROCESSED
```

```
C
C   IUSIT(N)              INDEX OF IN-PLACE SITE STORING UNIT SET N AT THE
C                         BEGINNING OF THE YEAR BEING PROCESSED
C
C   IWTMV(N)              TOTAL WEIGHT-KM (WEIGHT * KM) THAT UNIT SET N
C                         WAS MOVED IN AN ALLOCATION.
C
C   IX(K)                 ARRAY ORDERED, IN INCREASING ORDER, BY SUBRCUTINE
                          ORDER
C
C   JDIST(NS,MS)          DISTANCE, IN KM, BETWEEN SITE NS AND SITE MS
C
C   JFUL                  INDICATOR : SET = 1 WHEN A UNIT SET IS ALLOCATED
C                         TO A SITE THAT ALREADY HAS AT LEAST MXLIST UNIT SETS
C                         ALLOCATED TO IT
C
C   JKP(IG)               THE TOTAL AMOUNT (AREA) OF UNALLOCATED AREA REMAINING
C                         IN THE UNIT SETS OF PROJECT IG
C
C   JPRI(N)               INPUT UNIT SET PRIORITY OF UNIT SET N
C
C   JSIT(N)               SITE INDEX OF ALLOCATION SITE FOR UNIT SET N (READ
C                         FROM INPUT FILE 12 DURING ASSESSMENT PHASE (IND5 .GT.
                          0))
C                         OR SET TO INPUT IN-PLACE SITE DURING BASELINE CASE
C                         (IND5 .LT. 0))
C
C   JUN(N)                INDEX OF ALLOCATED UNIT SET N (READ FROM INPUT
C                         FILE12.TXT FOR ASSESSMENT PHASE (IND5 .GT. 0))
C                         OR SET TO INPUT IN-PLACE SITE DURING BASELINE
C                         CASE (IND5 .LT. 0) OR SET TO INPUT IN-PLACE SITE
C                         DURING ALLOCATION PHASE.
C
C   KPKG                  CONTROL PARAMETER ; KPKG .LE. 1 PUTS ALGORITHM IN THE
C                         'REDUCE PROJECT DISPERSION' OPTION.  KPKG .GE.  2
C                         PUTS ALGORITHM IN 'IGNORE PROJECT DISPERSION' OPTION
C
C   LABSI(NS)             NUMERIC SITE ID ASSOCIATED WITH SITE INDEX NS
C
C   LDIS(N)               THE DISTANCE OF THE NEAREST SITE TO THE GDP FOR
C                         UNIT SET WITH INDEX N
C
C   LLMURQ(N)             SUM OF STORAGE AREA AND WEIGHT FOR UNIT SET N
C                         IN THE PREVIOUS YEAR
C
C   LMUREQ(N)             SUM OF AREA AND WEIGHT FOR UNIT SET N IN THE YEAR
C                         BEING PROCESSED.
C
C   LTOT                  TOTAL WEIGHT (TONS) OF ALL UNIT SETS ALLOCATED IN
C                         A YEAR
C
C   MALL                  TOTAL AREA (SPACE REQUIREMENT) OF INPUT UNIT SETS
C                         (OVER ALL PROJECTS)
C
```

```
C   MFL                 NUMBER OF RECORDS IN THE DESIGNATED PROJECT/UNIT
C                       ALLOCATION INPUT FILE (FILE7.TXT)
C
C   MINP(NS,IG)         NUMBER OF UNIT SETS OF PROJECT IG WHICH ARE
C                       ALLOCATED TO SITE NS
C
C   MNUN                TOTAL NUMBER OF UNIT SETS ALLOCATED IN A YEAR
C
C   MOF(N)              INDICATOR OF WHETHER UNIT SET N HAS BEEN ALLOCATED.

C                       MOF(N) = 0 MEANS UNIT SET N IS ALLOCATED (OR

C                       UNAVAILABLE FOR ALLOCATION).  MOF(N) = 1 MEANS UNIT

C                       SET N IS UNALLOCATED AND AVAILABLE FOR ALLOCATION
C
C   MPREQ(IG)           TOTAL STORAGE AREA REQUIREMENT FOR ALL UNIT SETS
C                       IN PROJECT IG
C
C   MSIT                THE INDEX OF THE FICTITIOUS SITE -99 TO WHICH ALL
C                       UNIT SETS NOT FITTING ANYWHERE WILL BE ALLOCATED.
C
C   MTOT(IG)            TOTAL NUMBER OF UNIT SETS, OF NONZERO SIZE AND
                        WEIGHT,
C                       IN PROJECT IG.
C
C   MUREQ(N)            SPACE REQUIREMENT FOR UNIT SET N IN CURRENT YEAR
C
C   MUSIT(N)            IN-PLACE SITE FOR UNIT SET N DURING ASSESSMENT
C                       PHASE (AS READ FROM FILE12.TXT)
C
C   MUWT(N)             WEIGHT OF UNIT SET N IN CURRENT YEAR
C
C   MYR                 ID OF YEAR BEING PROCESSED. WHEN THE GOAL YEAR IS
C                       BEING PROCESSED INITIALLY (NYR = 1), ITS MYR ID IS
C                       MYR = 9.  FOR ALL OTHER YEARS PROCESSED, MYR IS
C                       THE YEAR ID READ FROM INPUT.
C
C   NALLU(NS,K)         UNIT INDEX OF K-TH ALLOCATED UNIT SET ALLOCATED
C                       TO SITE NS
C
C   NB                  TOTAL NUMBER OF BROKEN (DISPERSED OVER TWO OR
C                       MORE SITES) PROJECTS IN A YEAR
C
C   NCNT(IY)            TOTAL NUMBER OF UNIT SET ALLOCATIONS IN YEAR IY
C                       READ FROM FILE12.TXT AT START OF ASSESSMENT PHASE.
C
C   NDIS(N)             THE DISTANCE  UNIT N WAS MOVED FROM ITS IN PLACE SITE

C                       TO ITS ALLOCATED SITE (REDUNDANT)
C
C   NFIN                NUMBER OF YEARS SIMULATED. (IN THE ALLOCATION PHASE,
C                       THIS IS ONE MORE THAN THE ACTUAL NUMBER OF YEARS IN
                        THE
```

```
C                         SCENARIO BECAUSE THE GOAL YEAR IS RUN TWICE ( AT THE
C                         START OF PROCESSING AND AT THE END).
C
C   NIND(N)               THE RANK ORDER OF UNIT SET N AFTER THE UNIT SETS (OF

C                         A PROJECT) ARE RANKED BY INCREASING UNIT PRIORITY
C
C   NPAK                  NUMBER OF PROJECTS TO BE PROCESSED
C
C   NPRI(N)               COMBINED UNIT SET/PROJECT PRIORITY FOR UNIT SET N
C                         (COMPUTED FROM JPRI(N) AND IPRIO( ) )
C
C   NRP                   INDEX OF LAST PROJECT PROCESSED. (THIS ALWAYS = 25
C                         BECAUSE OF THE PRIORITY ORDERING)
C
C   NSCAP(NS)             SPACE CAPACITY (AREA) OF SITE WITH INDEX NS
C
C   NSIT                  THE NUMBER OF ACTUAL SITES (CUMULATED OVER YEARS,
C                         EVEN IF SOME ARE NOT OPEN IN SOME YEARS) PLUS 1 (THE
C                         CONUS SITE WHICH HAS INDEX NSIT). THIS EXCLUDES
C                         THE OVERFLOW SITE -99 (WHICH HAS INDEX (NSIT + 1)
C
C
C   NSPOC(NS)             AMOUNT OF SPACE ALLOCATED AT SITE WITH INDEX NS
C
C   NTOX(N)               TOTAL NUMBER OF OPEN SITES IN THE ORDERED SITE
C                         PREFERENCE LIST FOR PROJECT N, WHERE SITE
C                         PREFERENCE IS BASED ON AVERAGE CLOSENESS OF
C                         PROJECT N'S UNIT SET GDPS TO A SITE.
C
C   NTUN                  TOTAL NUMBER OF INPUT UNIT SETS BEING PROCESSED.
C
C   NUCP(IG)             NR UNIT SETS OF PROJECT IG THAT WERE RESITED AND WERE
C                        PRESENT IN PREVIOUS YEAR
C
C   NUIS(NS)             NUMBER OF UNIT SETS IN THE INITIAL IN-PLACE LIST FOR
C                        SITE NS (OVER ALL UNIT SETS AND PROJECTS)
C
C   NUMP                 TOTAL NUMBER OF UNIT SETS ALLOCATED THIS YEAR
C
C   NUNP(IG)              NUMBER OF UNIT SETS IN PROJECT IG
C
C   NYR                   INDEX OF THE YEAR BEING PROCESSED (NOTE : NYR = MYR
                          + 1
C                         FOR ALL NYR .GT. 1)
C
C   XDG                   TOTAL SUMMED TON*KM FROM UNIT SET ALLOCATION SITE TO
C                         UNIT SET GDP OVER ALL UNIT SETS ALLOCATED IN A YEAR
C
C   ZNUC(NY,IG)           FRACTION OF ALLOCATED UNIT SETS OF PROJECT IG
C                         THAT WERE RESITED IN YEAR NY
C
C   ZTP                   TOTAL NUMBER OF NON-EMPTY PROJECTS ALLOCATED
C                         (OR ASSESSED) IN A YEAR
```

```
C
C
C   LOCAL ARRAYS AND INDICATORS : MAIN PROGRAM
C
C
C   DLAT(NS)                  THE LATITUDE, IN RADIANS NORTH, OF SITE WITH INDEX NS
C
C   DLON(NS)                  THE LONGITUDE, IN RADIANS EAST OF GREENWICH, OF THE
C                             SITE WITH INDEX NS
C
C   IIPUN(N)                  THE INDEX OF THE PROJECT CONTAINING THE UNIT SET
C                             WITH INDEX N IN THE PREVIOUS YEAR (USED TO TRANSFER
C                             ASSOCIATED PROJECT INDEX TO SAME UNIT WITH NEW INDEX
C                             IN THE YEAR BEING PROCESSED.)
C
C   IIU(N)                    THE UNIT SET NUMBER OF THE UNIT SET WITH INDEX N
C                             IN THE PREVIOUS YEAR (USED TO TRANSFER THE UNIT SET
C                             NUMBER TO SAME UNIT WITH NEW INDEX IN THE YEAR BEING
C                             PROCESSED.)
C
C   INPRI(N)                  THE INTERNAL UNIT SET PRIORITY OF THE UNIT SET
C                             WITH INDEX N IN THE PREVIOUS YEAR (USED TO TRANSFER
C                             PRIORITY TO SAME UNIT WITH NEW INDEX IN THE YEAR
C                             BEING PROCESSED.)
C
C   ISPR(NS)                  INPUT RETENTION PRIORITY OF SITE NS.  USED TO ORDER
C                             SITES FOR DETERMINATION OF REDUNDANT SITES. SITING
                              SETS
C                             ISPR(NS) =  -(SITE NS CAPACITY) BY DEFAULT SO THAT
C                             LARGER SITES ARE PREFERRED FOR RETENTION.
C
C   ITRANS(NS)                THE INDEX, IN THE YEAR BEING PROCESSED, OF THE SITE
C                             WITH INDEX NS IN THE PREVIOUS YEAR.
C
C   IWW(NS)                   THE UNWEIGHTED TRANSHIPMENT COMPONENT OF THE AVERAGE
C                             UNIT SET MOE OVER ALL ALLOCATED UNITS OF A PROJECT
C
C   LLABSI(NS)                NUMERIC ID OF SITE NS IN THE PREVIOUS YEAR (RELATIVE
C                             TO THE YEAR BEING PROCESSED)
C
C   LSIT                      THE FICTITIOUS SITE -01 USED TO AS IN-PLACE SITE
C                             ASSIGNED TO NEW UNITS 'COMING FROM CONUS' AND ALSO
C                             USED AS AN OVERFLOW STORAGE SITE WHEN SITE-99 IS FULL
C                             (I.E. HAS MXLIST UNITS ALLOCATED TO IT).
C
C   NNT(NS)                   NUMBER OF UNIT SETS IN-PLACE AT SITE NS.
```

GLOSSARY

1. ABBREVIATIONS, ACRONYMS, AND SHORT TERMS

avg	average
CAA	US Army Concepts Analysis Agency
CONUS	continental United States
DOS	disk operating system
GC	Great Circle
GDP	general defense plan
ID	identification
K	thousands (of bytes)
km	kilometer(s)
MOE	measure(s) of effectiveness
PC	personal computer
proj	project
OPLAN	operation plan
POMCUS	prepositioned materiel configured to unit sets
POMCUSITE	POMCUS Unit Siting Alternatives (study)
RAM	random acccess memory
sq	square
STON	short ton(s)
US	unit set(s)
USAREUR	United States Army Europe
wt	weight
yr	year

2. DEFINITIONS

GDP coordinates
Longitude and latitude of unit assembly or staging area

project
Group of units belonging to a combat formation or in support of a specific combat formation

CAA	**ALGORITHM FOR SITING POMCUS EQUIPMENT AT STORAGE FACILITIES (SITING)**	**STUDY SUMMARY CAA-TP-91-3**

THE REASON FOR PERFORMING THE STUDY was to provide a baseline technical reference on the development of the allocation model denoted as SITING. The SITING model was developed in the POMCUSITE (POMCUS Unit Siting Alternatives) Study at the US Army Concepts Analysis Agency (CAA). SITING was designed to assist in POMCUS (prepositioning of materiel configured to unit sets) program management. This technical reference is needed to explain the full analytic nature and capabilities of the SITING Model, including features not used in the POMCUSITE Study.

THE STUDY SPONSOR is the Director, US Army Concepts Analysis Agency.

THE STUDY OBJECTIVES WERE TO document the analytic basis of the SITING Model, including inputs, outputs, and operation.

THE SCOPE OF THE STUDY addresses the efficient allocation of sets of equipment to storage sites in a POMCUS context over a timeframe of up to 8 consecutive years.

THE MAIN ASSUMPTIONS OF THIS WORK are:

(1) The use of Great Circle (GC) distances between allocation sites is appropriate for model applications.

(2) The penalty for moving a unit set between locations can be treated as proportional to the product of the unit set weight and the distance moved.

(3) Unit sets will not be fractionated (broken into pieces) in actual allocations.

(4) All (latitude, longitude) locations are in the northern hemisphere.

THE BASIC APPROACHES USED IN THIS STUDY were to:

(1) Define the allocation problem SITING is designed to solve.

(2) Describe the analytic nature of the allocation algorithm.

(3) Define the inputs and outputs of the model.

(4) Describe model operation and use.

THE PRINCIPAL FINDINGS of the work reported herein are:

(1) SITING is a deterministic allocation model written in FORTRAN for an IBM personal computer (PC). The model is designed to allocate sets of equipment (unit sets) efficiently over a set of storage sites.

(2) SITING allocates unit sets to storage sites while seeking to meet combinations of the following objectives:

(a) Each unit set is stored close to a predesignated (for that unit) location.

(b) The likelihood of an allocated unit set having to change storage site from one year to the next is kept small.

(c) Predesignated groupings of unit sets are usually allocated to a single storage site.

(3) The degree to which SITING achieves each of the above objectives is partly a function of user-specified options for SITING operation.

THE STUDY EFFORT was directed by Mr. Walter J. Bauman, Force Systems Directorate, US Army Concepts Analysis Agency.

COMMENTS AND QUESTIONS may be sent to the Director, US Army Concepts Analysis Agency, ATTN: CSCA-FSC, 8120 Woodmont Avenue, Bethesda, Maryland 20814-2797.

CAA	**ALGORITHM FOR SITING POMCUS EQUIPMENT AT STORAGE FACILITIES (SITING)**	**STUDY SUMMARY CAA-TP-91-3**

THE REASON FOR PERFORMING THE STUDY was to provide a baseline technical reference on the development of the allocation model denoted as SITING. The SITING model was developed in the POMCUSITE (POMCUS Unit Siting Alternatives) Study at the US Army Concepts Analysis Agency (CAA). SITING was designed to assist in POMCUS (prepositioning of materiel configured to unit sets) program management. This technical reference is needed to explain the full analytic nature and capabilities of the SITING Model, including features not used in the POMCUSITE Study.

THE STUDY SPONSOR is the Director, US Army Concepts Analysis Agency.

THE STUDY OBJECTIVES WERE TO document the analytic basis of the SITING Model, including inputs, outputs, and operation.

THE SCOPE OF THE STUDY addresses the efficient allocation of sets of equipment to storage sites in a POMCUS context over a timeframe of up to 8 consecutive years.

THE MAIN ASSUMPTIONS OF THIS WORK are:

(1) The use of Great Circle (GC) distances between allocation sites is appropriate for model applications.

(2) The penalty for moving a unit set between locations can be treated as proportional to the product of the unit set weight and the distance moved.

(3) Unit sets will not be fractionated (broken into pieces) in actual allocations.

(4) All (latitude, longitude) locations are in the northern hemisphere.

THE BASIC APPROACHES USED IN THIS STUDY were to:

(1) Define the allocation problem SITING is designed to solve.

(2) Describe the analytic nature of the allocation algorithm.

(3) Define the inputs and outputs of the model.

(4) Describe model operation and use.

THE PRINCIPAL FINDINGS of the work reported herein are:

(1) SITING is a deterministic allocation model written in FORTRAN for an IBM personal computer (PC). The model is designed to allocate sets of equipment (unit sets) efficiently over a set of storage sites.

(2) SITING allocates unit sets to storage sites while seeking to meet combinations of the following objectives:

(a) Each unit set is stored close to a predesignated (for that unit) location.

(b) The likelihood of an allocated unit set having to change storage site from one year to the next is kept small.

(c) Predesignated groupings of unit sets are usually allocated to a single storage site.

(3) The degree to which SITING achieves each of the above objectives is partly a function of user-specified options for SITING operation.

THE STUDY EFFORT was directed by Mr. Walter J. Bauman, Force Systems Directorate, US Army Concepts Analysis Agency.

COMMENTS AND QUESTIONS may be sent to the Director, US Army Concepts Analysis Agency, ATTN: CSCA-FSC, 8120 Woodmont Avenue, Bethesda, Maryland 20814-2797.

ALGORITHM FOR SITING POMCUS EQUIPMENT AT STORAGE FACILITIES (SITING)	STUDY SUMMARY CAA-TP-91-3

THE REASON FOR PERFORMING THE STUDY was to provide a baseline technical reference on the development of the allocation model denoted as SITING. The SITING model was developed in the POMCUSITE (POMCUS Unit Siting Alternatives) Study at the US Army Concepts Analysis Agency (CAA). SITING was designed to assist in POMCUS (prepositioning of materiel configured to unit sets) program management. This technical reference is needed to explain the full analytic nature and capabilities of the SITING Model, including features not used in the POMCUSITE Study.

THE STUDY SPONSOR is the Director, US Army Concepts Analysis Agency.

THE STUDY OBJECTIVES WERE TO document the analytic basis of the SITING Model, including inputs, outputs, and operation.

THE SCOPE OF THE STUDY addresses the efficient allocation of sets of equipment to storage sites in a POMCUS context over a timeframe of up to 8 consecutive years.

THE MAIN ASSUMPTIONS OF THIS WORK are:

(1) The use of Great Circle (GC) distances between allocation sites is appropriate for model applications.

(2) The penalty for moving a unit set between locations can be treated as proportional to the product of the unit set weight and the distance moved.

(3) Unit sets will not be fractionated (broken into pieces) in actual allocations.

(4) All (latitude, longitude) locations are in the northern hemisphere.

THE BASIC APPROACHES USED IN THIS STUDY were to:

(1) Define the allocation problem SITING is designed to solve.

(2) Describe the analytic nature of the allocation algorithm.

(3) Define the inputs and outputs of the model.

(4) Describe model operation and use.

THE PRINCIPAL FINDINGS of the work reported herein are:

(1) SITING is a deterministic allocation model written in FORTRAN for an IBM personal computer (PC). The model is designed to allocate sets of equipment (unit sets) efficiently over a set of storage sites.

(2) SITING allocates unit sets to storage sites while seeking to meet combinations of the following objectives:

(a) Each unit set is stored close to a predesignated (for that unit) location.

(b) The likelihood of an allocated unit set having to change storage site from one year to the next is kept small.

(c) Predesignated groupings of unit sets are usually allocated to a single storage site.

(3) The degree to which SITING achieves each of the above objectives is partly a function of user-specified options for SITING operation.

THE STUDY EFFORT was directed by Mr. Walter J. Bauman, Force Systems Directorate, US Army Concepts Analysis Agency.

COMMENTS AND QUESTIONS may be sent to the Director, US Army Concepts Analysis Agency, ATTN: CSCA-FSC, 8120 Woodmont Avenue, Bethesda, Maryland 20814-2797.